Pentesting Methodologies and Tools

Web and Internet Security

UBB Cyber Security Master's Program

April 14th, 2025

Tudor Damian

- **Cloud & Cybersecurity Advisor** @ D3 Cyber
  - Cybersecurity & vCISO Services
  - Cloud Strategy & Governance
  - IT Risk Management
  - Business Process Optimization
- **Guild Master** @ The Guild Hall

- **Contact**: tudor.damian@d3cyber.eu | tudy.ro

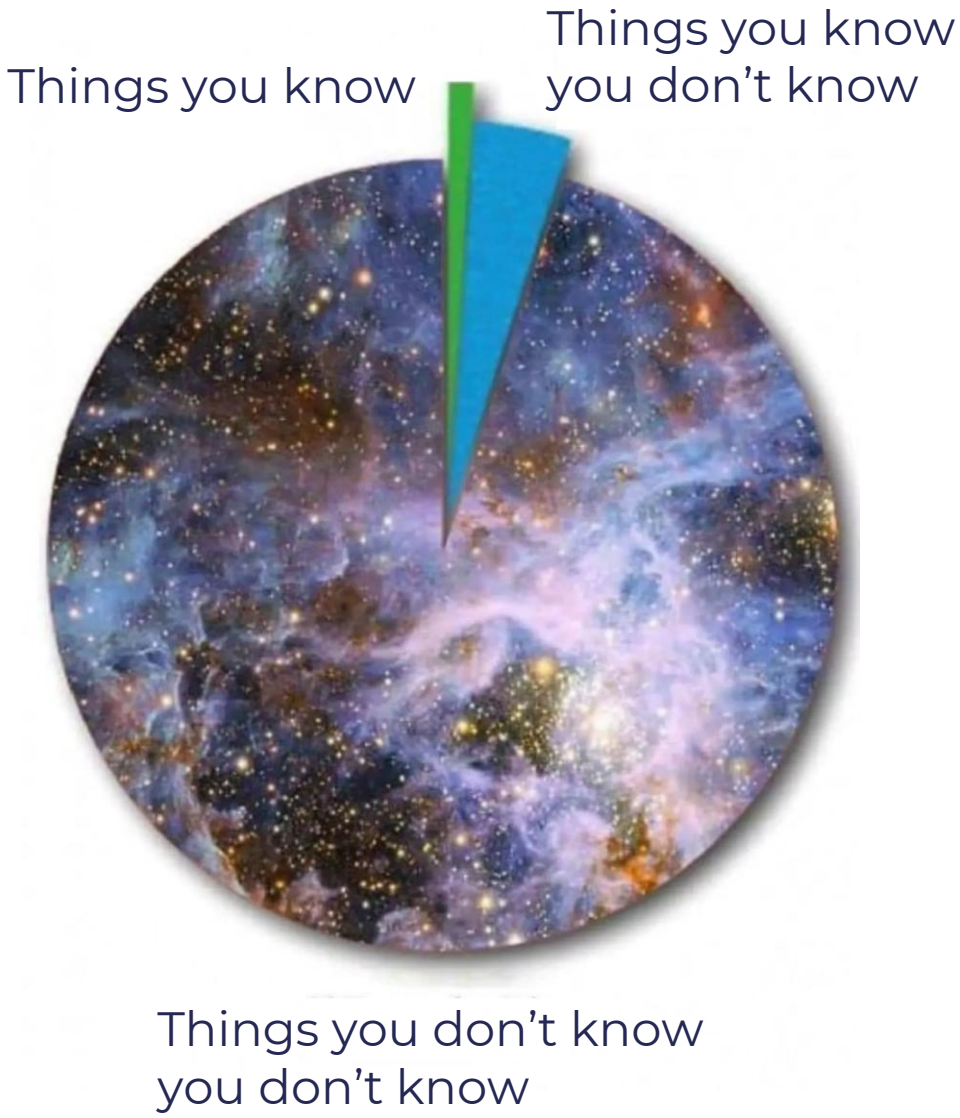
Pentesting Methodologies and Tools

- Cybersecurity Fundamentals
- Anatomy of an Attack
- Security Assessments, Audits & Pentests
- Pentesting Methodologies
- Pentesting Tools
- Self Assessments

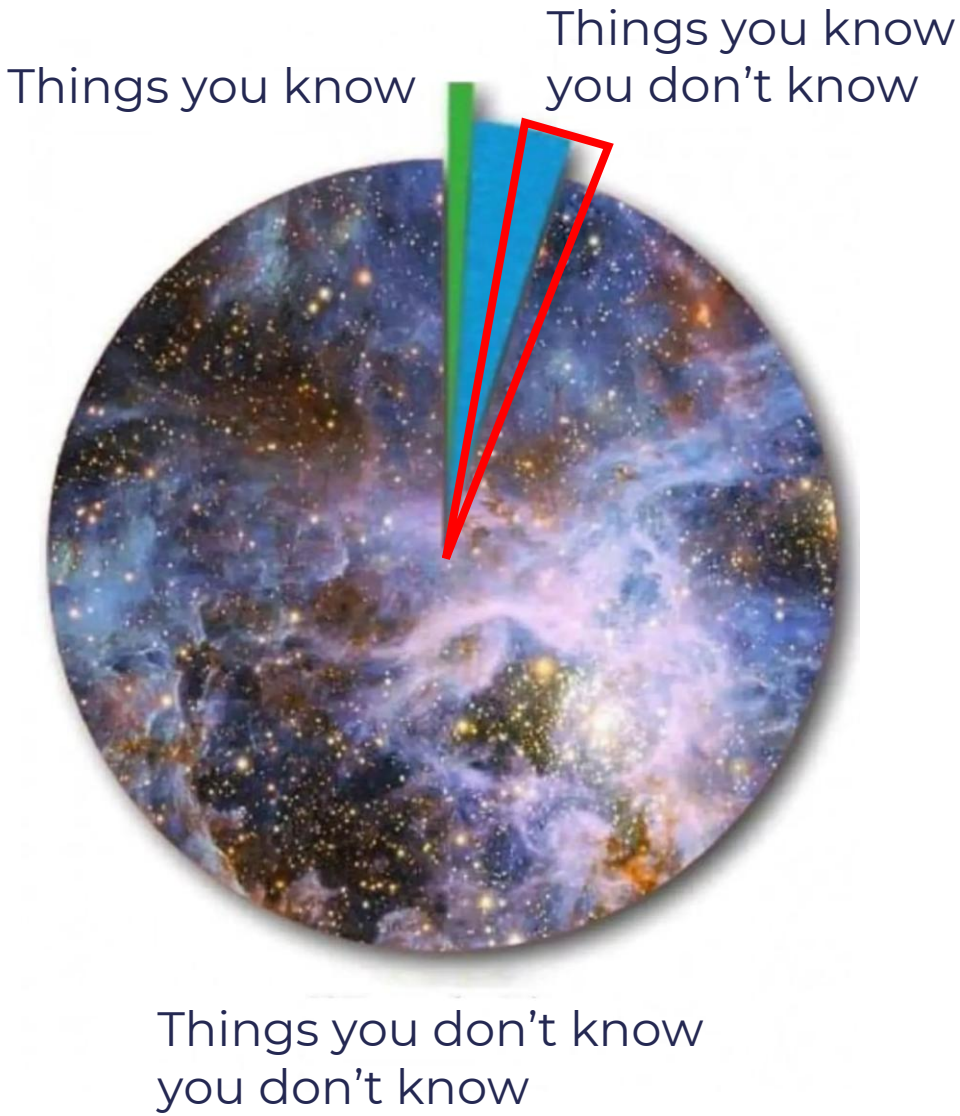
Got questions? Ask away!

- Ask questions at any time – **there's no Q&A at the end!**
- Yes, you'll get the slides! ☺

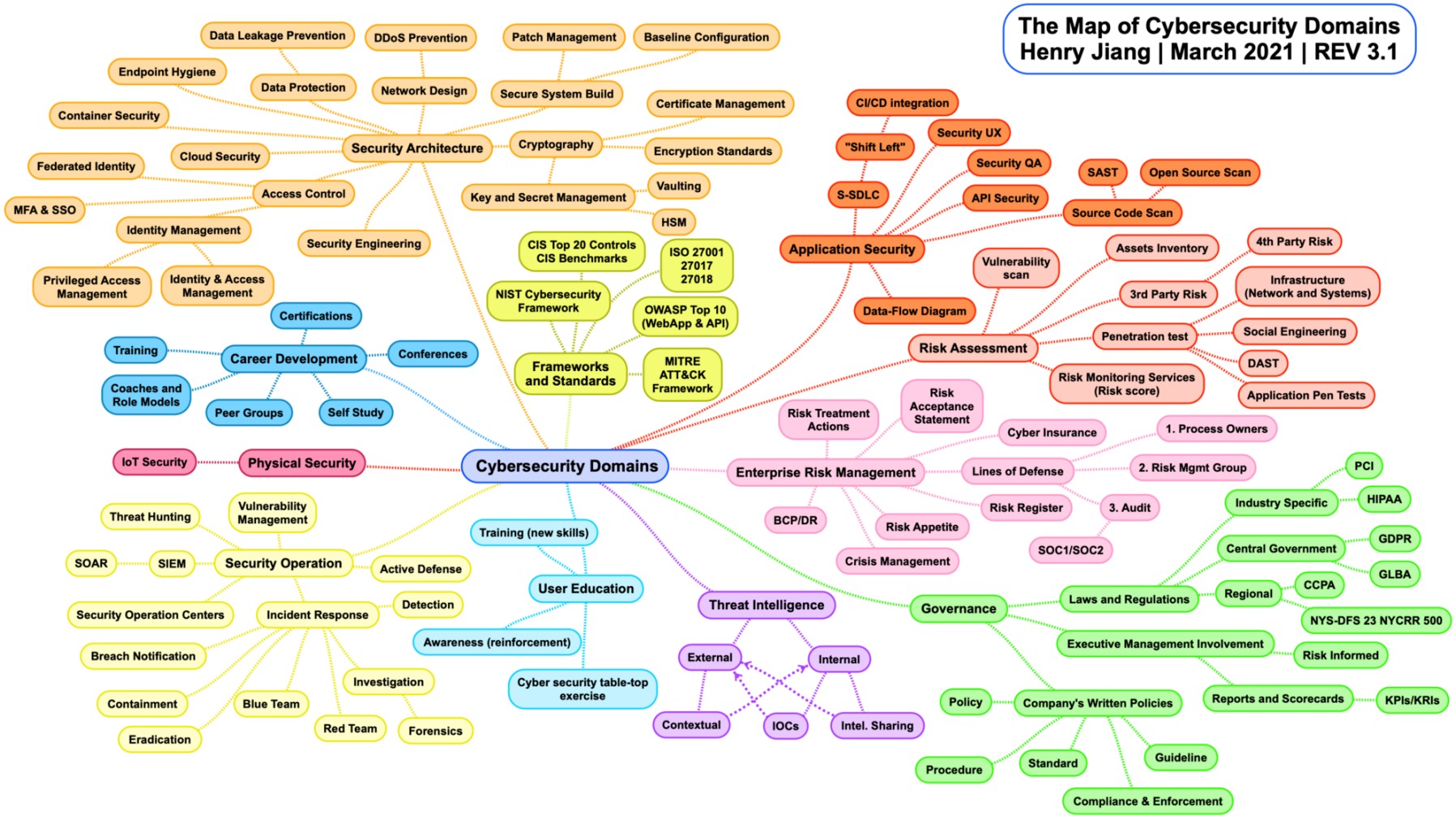
This is an overview talk ☺



This is an overview talk ☺



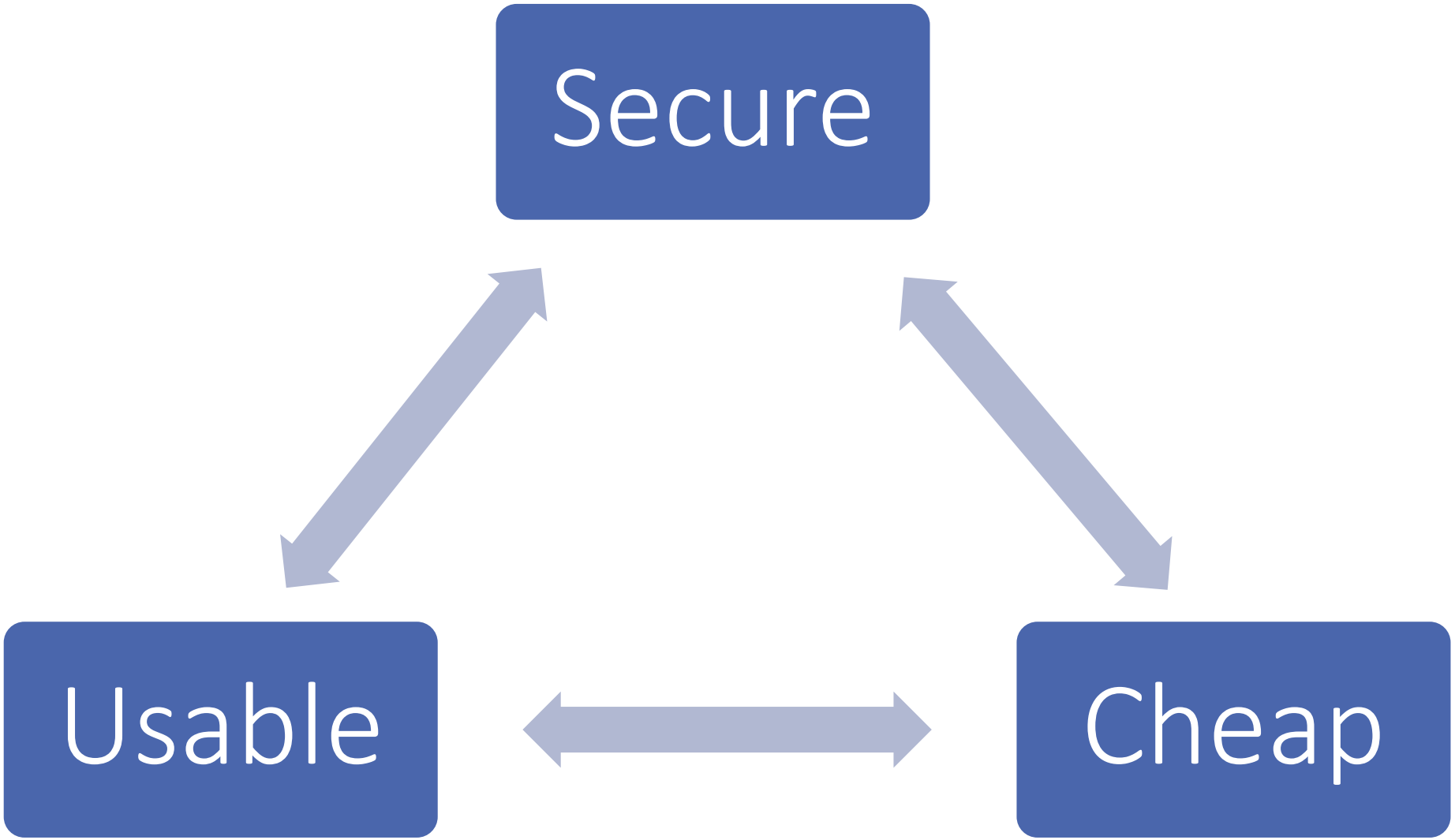
Cybersecurity Domains Map 3.1



Foundations of Security

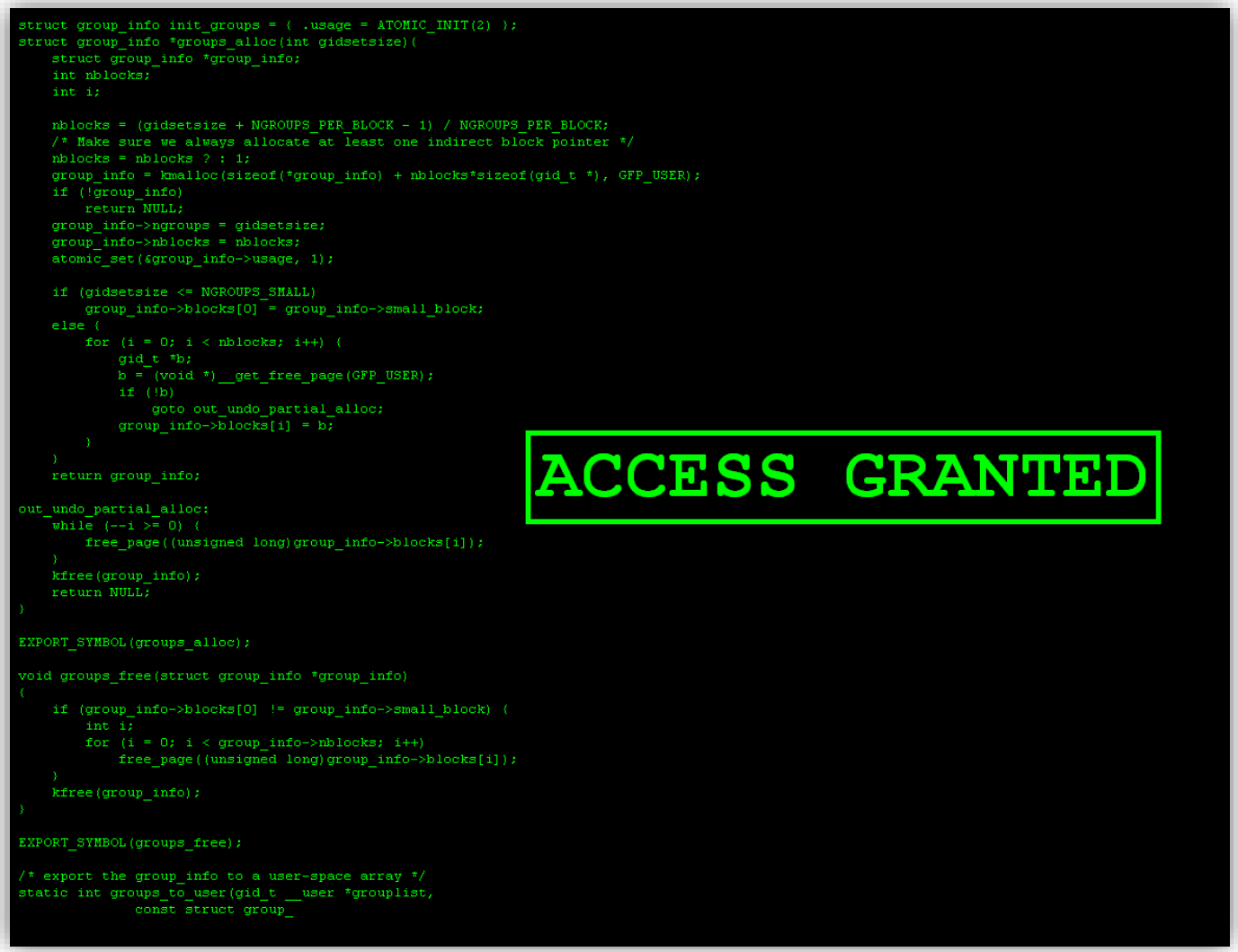
Your systems security is as strong as its weakest link

Pick any two! ☺



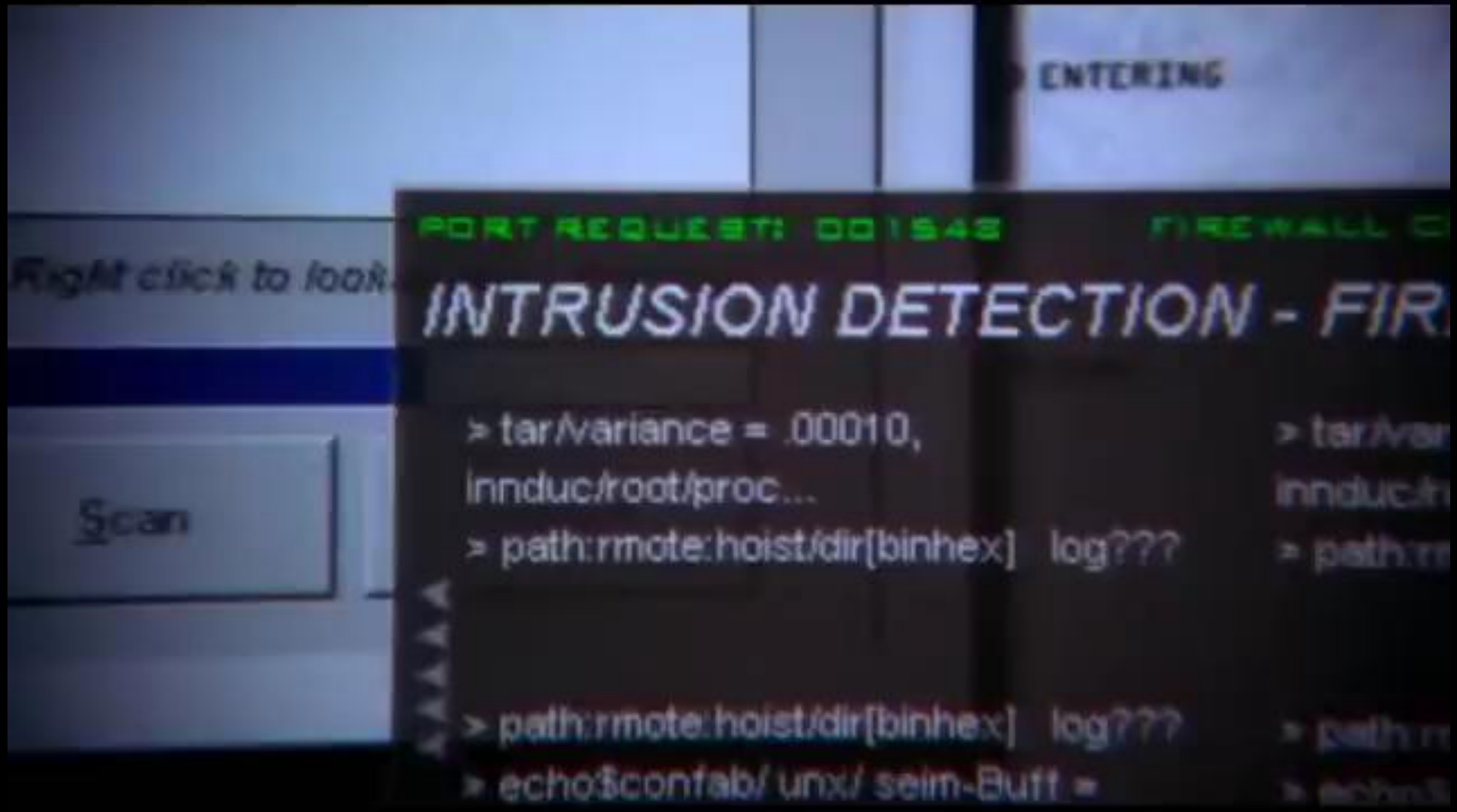
Cybersecurity Fundamentals

There’s lots of fake information out there




```
struct group_info init_groups = { .usage = ATOMIC_INIT(2) };
struct group_info *groups_alloc(int gidsetsize){
    struct group_info *group_info;
    int nblocks;
    int i;

    nblocks = (gidsetsize + NGROUPS_PER_BLOCK - 1) / NGROUPS_PER_BLOCK;
    /* Make sure we always allocate at least one indirect block pointer */
    nblocks = nblocks ? : 1;
    group_info = kmalloc(sizeof(*group_info) + nblocks*sizeof(gid_t *), GFP_USER);
    if (!group_info)
        return NULL;
    group_info->ngroups = gidsetsize;
    group_info->nblocks = nblocks;
    atomic_set(&group_info->usage, 1);

    if (gidsetsize <= NGROUPS_SMALL)
        group_info->blocks[0] = group_info->small_block;
    else {
        for (i = 0; i < nblocks; i++) {
            gid_t *b;
            b = (void *)__get_free_page(GFP_USER);
            if (!b)
                goto out_undo_partial_alloc;
            group_info->blocks[i] = b;
        }
    }
    return group_info;

out_undo_partial_alloc:
    while (--i >= 0) {
        free_page((unsigned long)group_info->blocks[i]);
    }
    kfree(group_info);
    return NULL;
}

EXPORT_SYMBOL(groups_alloc);

void groups_free(struct group_info *group_info)
{
    if (group_info->blocks[0] != group_info->small_block) {
        int i;
        for (i = 0; i < group_info->nblocks; i++)
            free_page((unsigned long)group_info->blocks[i]);
    }
    kfree(group_info);
}

EXPORT_SYMBOL(groups_free);

/* export the group_info to a user-space array */
static int groups_to_user(gid_t __user *grouplist,
            const struct group_
```

https://youtu.be/u8qgehH3kEQ



Elements of Information Security

- A state of well-being of information and infrastructure in which the possibility of theft, tampering, and disruption of information and services is kept low or tolerable

Confidentiality	**Integrity**	**Availability**	**Authenticity**
Non-repudiation	**Authorization**	**Accountability**	**Auditability**

Elements of Information Security

Component	Definition	Importance	Implementation
Confidentiality	Ensures that information is accessible only to those authorized to have access.	Protects sensitive data from unauthorized disclosure, maintaining privacy and preventing data breaches.	Encryption, access controls, and secure authentication mechanisms.
Integrity	Assures that the information is trustworthy and unaltered during transmission or storage.	Maintains the accuracy and consistency of data over its entire lifecycle.	Hash functions, checksums, digital signatures, and version control systems.
Availability	Ensures that data remains available to authorized users when needed.	Critical for business continuity and operational effectiveness.	Redundancy, failover strategies, regular maintenance, and protection against Denial-of-Service (DoS) attacks.
Authentication	Verifies the identity of the parties involved in communication.	Prevents unauthorized access by ensuring that users are who they claim to be.	Passwords, biometrics, multi-factor authentication (MFA), and digital certificates.

Elements of Information Security

Component	Definition	Importance	Implementation
Non-repudiation	Prevents an entity from denying the authenticity of their signature on a document or the sending of a message that they originated.	Provides legal proof and accountability, essential in transactions and communications.	Digital signatures, audit logs, and secure transaction records.
Authorization	Determines what an authenticated user is allowed to do; the process of granting or denying access rights and privileges to resources.	Ensures that users can only access resources and perform actions they are permitted to, preventing unauthorized activities.	Access control lists (ACLs), role-based access control (RBAC), permissions settings, and security policies.
Accountability	Ensures that actions of an entity can be traced uniquely to that entity, which can be held responsible for its actions.	Promotes responsible behavior by holding individuals accountable for their actions, essential for compliance and forensic analysis.	Audit trails, user activity logs, monitoring systems, and unique user identifiers.
Auditability	The capability to conduct a thorough review and examination of system records and activities to assess compliance, performance, or detect anomalies.	Facilitates detection of security incidents, ensures adherence to policies and regulations, and helps in optimizing system performance.	Logging mechanisms, regular security audits, compliance checks, and audit software tools.

Information Security Threats (1)

Natural Threats	Physical Security Threats	Human threats
• Natural disasters • Floods • Earthquakes • Hurricanes	• Loss or damage of system resources • Physical intrusion • Sabotage, espionage and errors	• Hackers • Insiders • Social engineering • Lack of knowledge and awareness

Information Security Threats (2)

Network Threats	Host Threats	Application Threats
• Information gathering • Sniffing and eavesdropping • Spoofing • Session hijacking • Man-in-the-middle attacks • ARP Poisoning • Password-based attacks • Denial of service attack • Compromised-key attack	• Malware attacks • Target Footprinting • Password attacks • Denial of service attacks • Arbitrary code execution • Unauthorized access • Privilege escalation • Backdoor Attacks • Physical security threats	• Data/input validation • SQL injection • Authentication and Authorization attacks • Configuration management • Information disclosure • Session management issues • Buffer overflow issues • Cryptography attacks • Parameter manipulation • Improper error handling and exception management • Auditing and logging issues

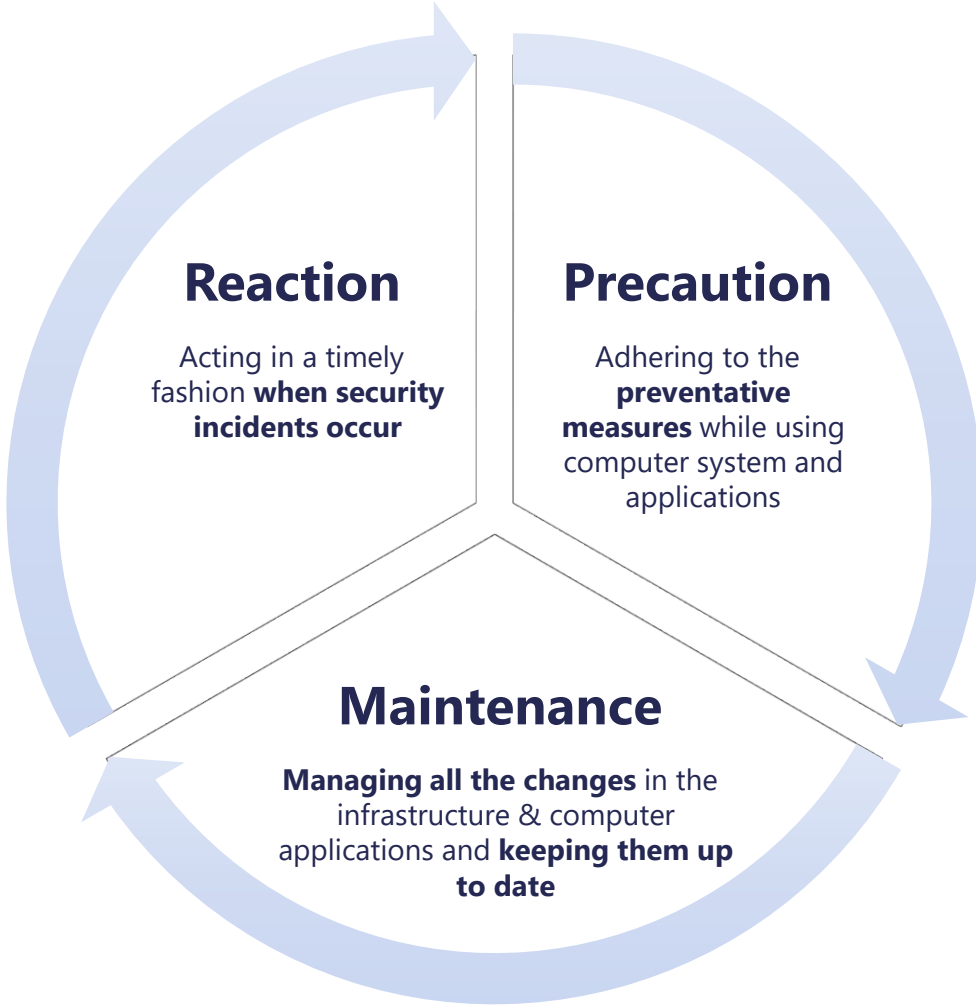
Exploiting vulnerabilities

"...as soon as a vulnerability is discovered, approximately 58% of ethical hackers can penetrate an environment in less than five hours."

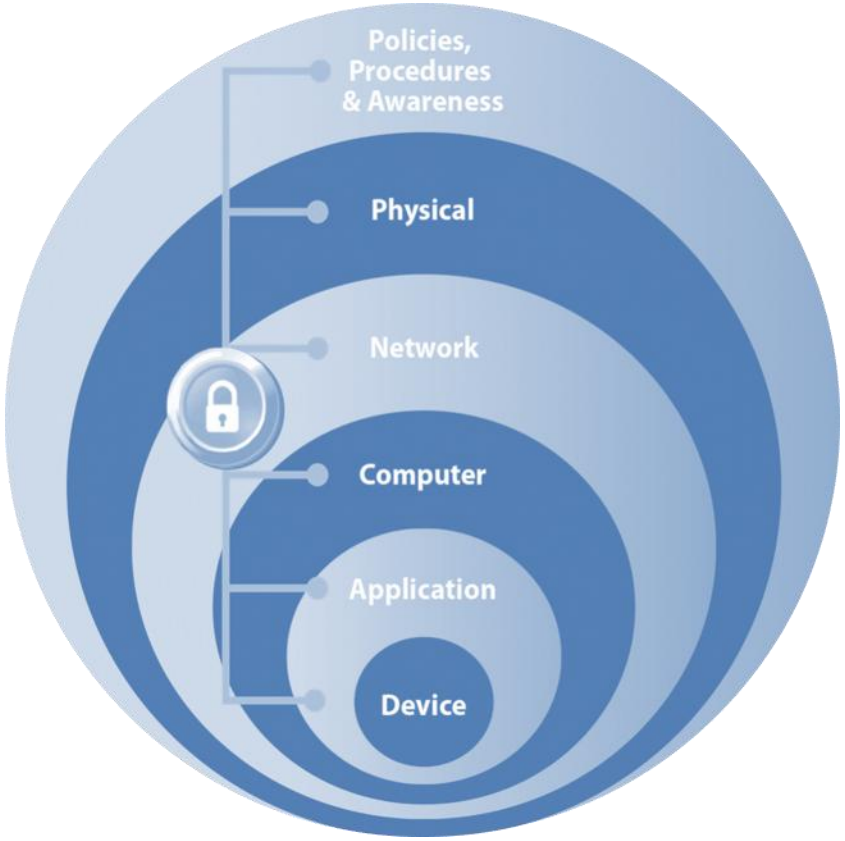
Remediation of vulnerabilities

- Only **20%** of all findings are remediated in **less than 30 days**
- For **80%** of all findings, it takes **30 days or longer** to be resolved
- For **57%** of all findings, it takes **90 days or longer** to be resolved
- **On average,** it takes 215 days

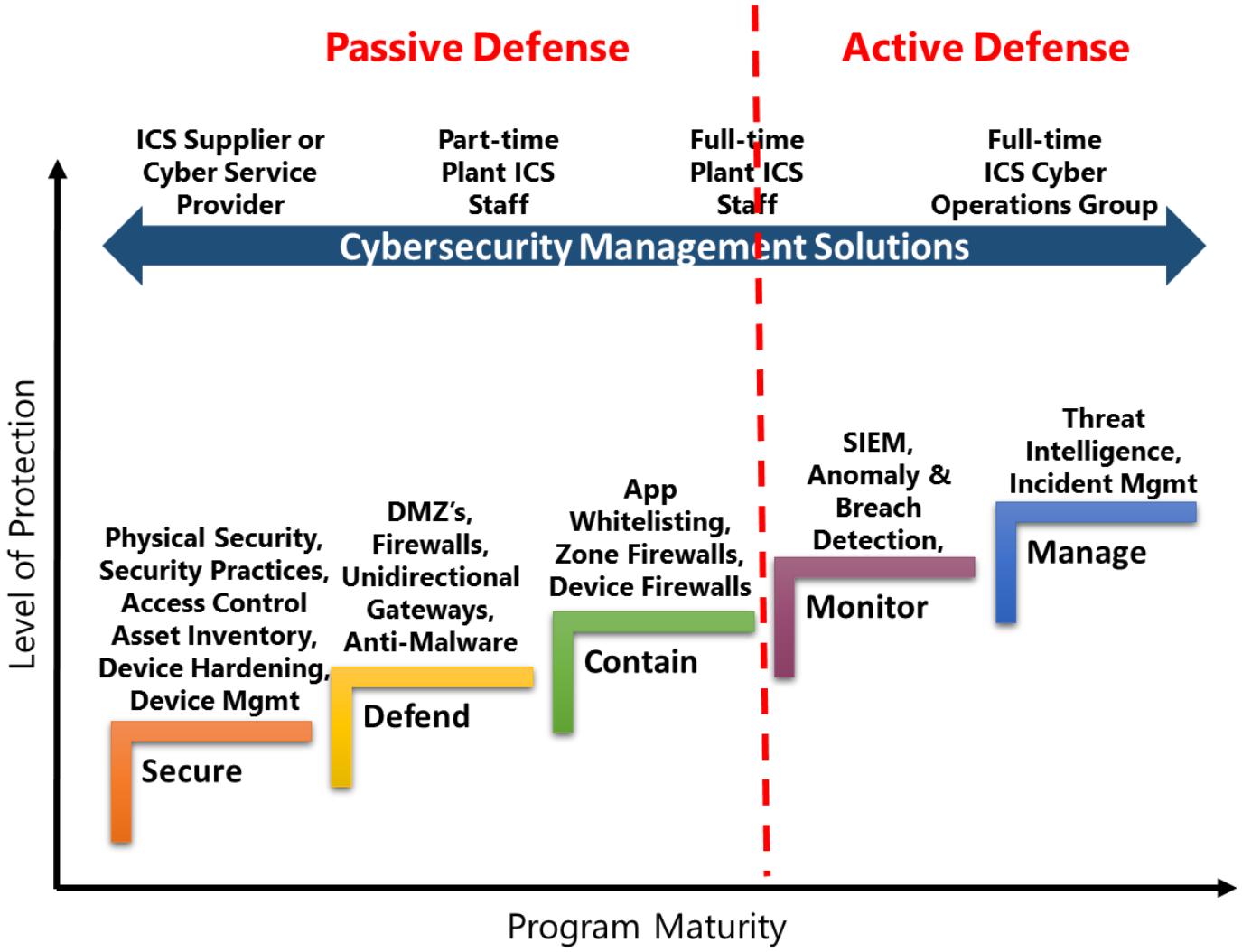
Fundamental Concepts of Security



Defense in Depth



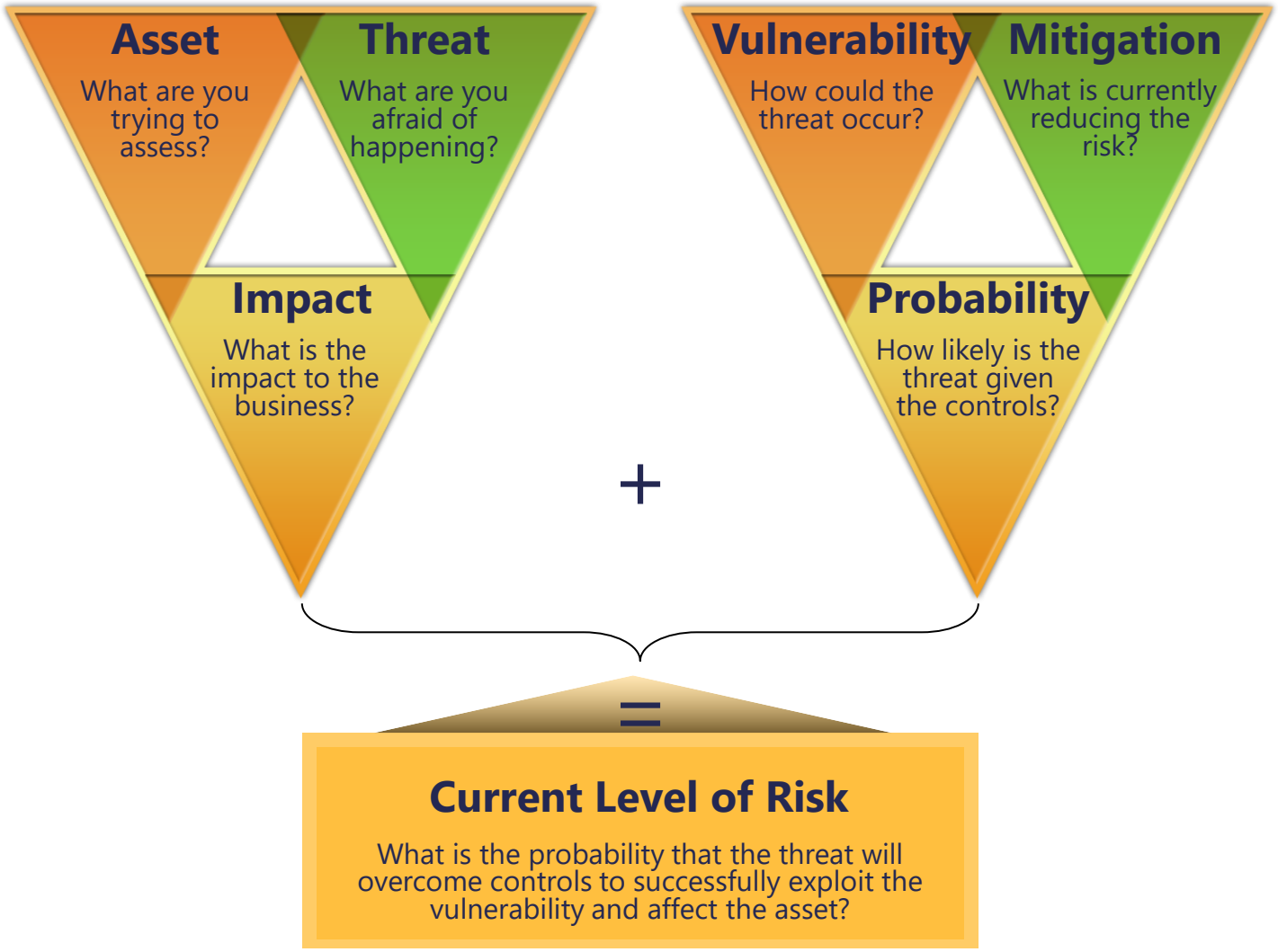
Defending against attacks



GRC – (how) are you doing it today?

- **Governance**, **Risk Management** & **Compliance** (**GRC**) are three facets that help to ensure that an organization meets its objectives
- **Goals:**
  - Keeping risk at acceptable levels
  - Maintaining availability to systems and services
  - Complying with relevant laws and regulations
  - Protecting customer and internal data

Risk Assessment & Risk Management



Example: COBIT

GRC – items in focus

- **Regulatory compliance** (e.g. PCI-DSS, HIPAA, CDSA, MPAA, etc.)
- **Data governance** (e.g. DLP, encrypting PII, geo location, etc.)
- **Financial governance** (e.g. CAPEX vs OPEX, prediction, cost centers, etc.)
- **Change management** (e.g. DevOps, user & organization readiness, etc.)
- **Business & market changes** and challenges
- **ITIL & COBIT**
  - Strategy, Design, Transition, Operation & Improvement
  - Ensure clear ownership & responsibilities
  - Better manage IT investments
  - Identify & handle IT risk

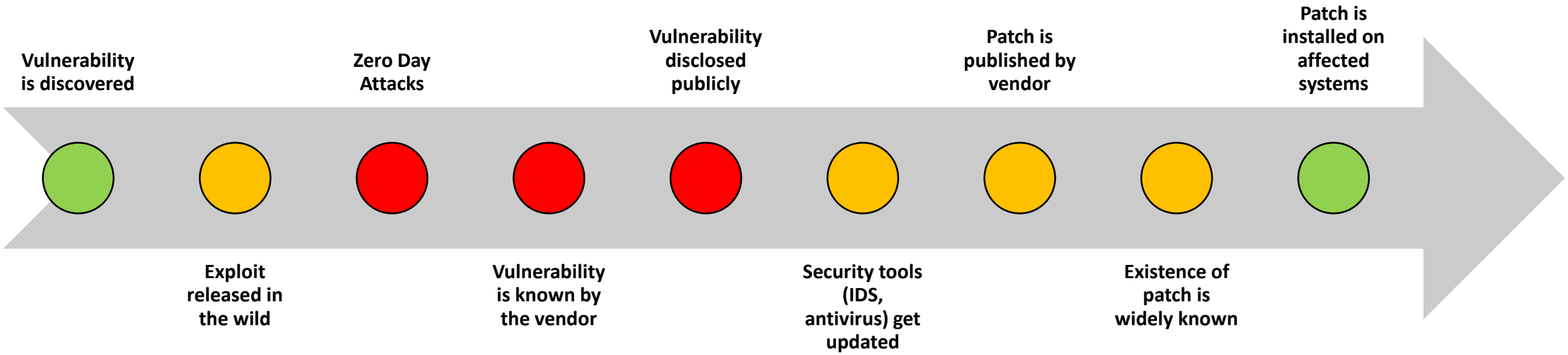
Understanding cloud security controls

On-premises	IaaS	PaaS	SaaS
1. Security Strategy, Governance, and Operationalization: Provide clear vision, standards and guidance for the company			
2. Administrative Control: Defend against loss of control of your Cloud services and on-premises systems			
3. Data: Identify and protect your most important information assets			
4. User Identity and Device Security: Strengthen protection for accounts and devices			
5. Application Security: Ensure application code is resilient to attacks			
6. Network: Ensure connectivity, isolation, and visibility into anomalous attacks			
7. OS and Middleware: Protect integrity of (virtual) hosts			
8. On-prem / private environments: Secure the foundation			

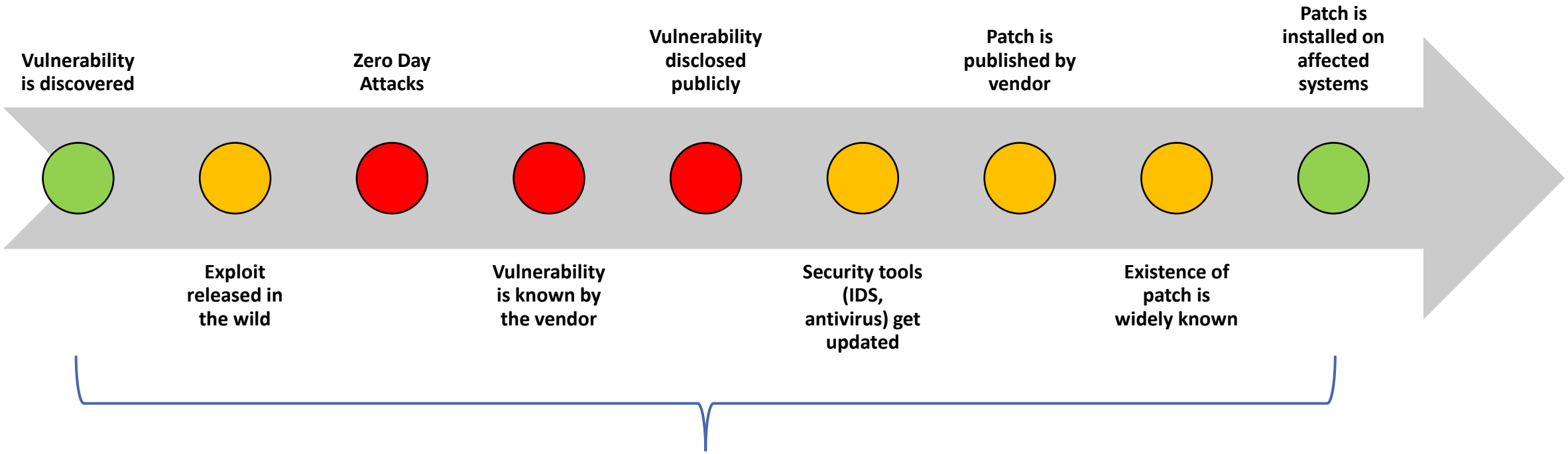
There's a lot of stuff to protect

Anatomy of an attack

Window of Vulnerability - overview

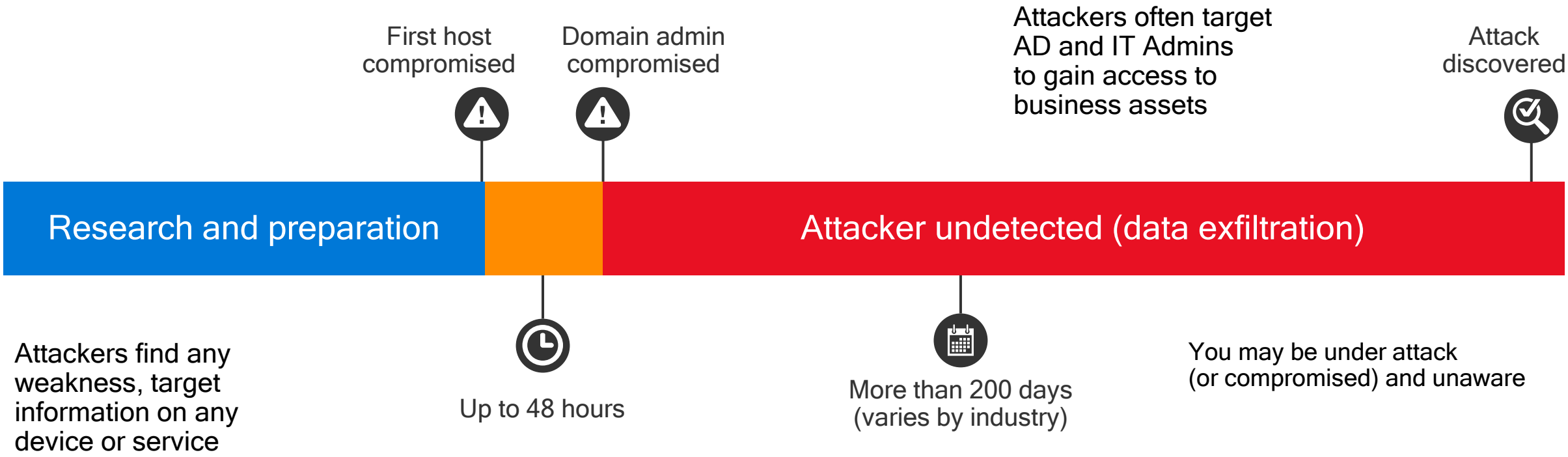


Window of Vulnerability - overview



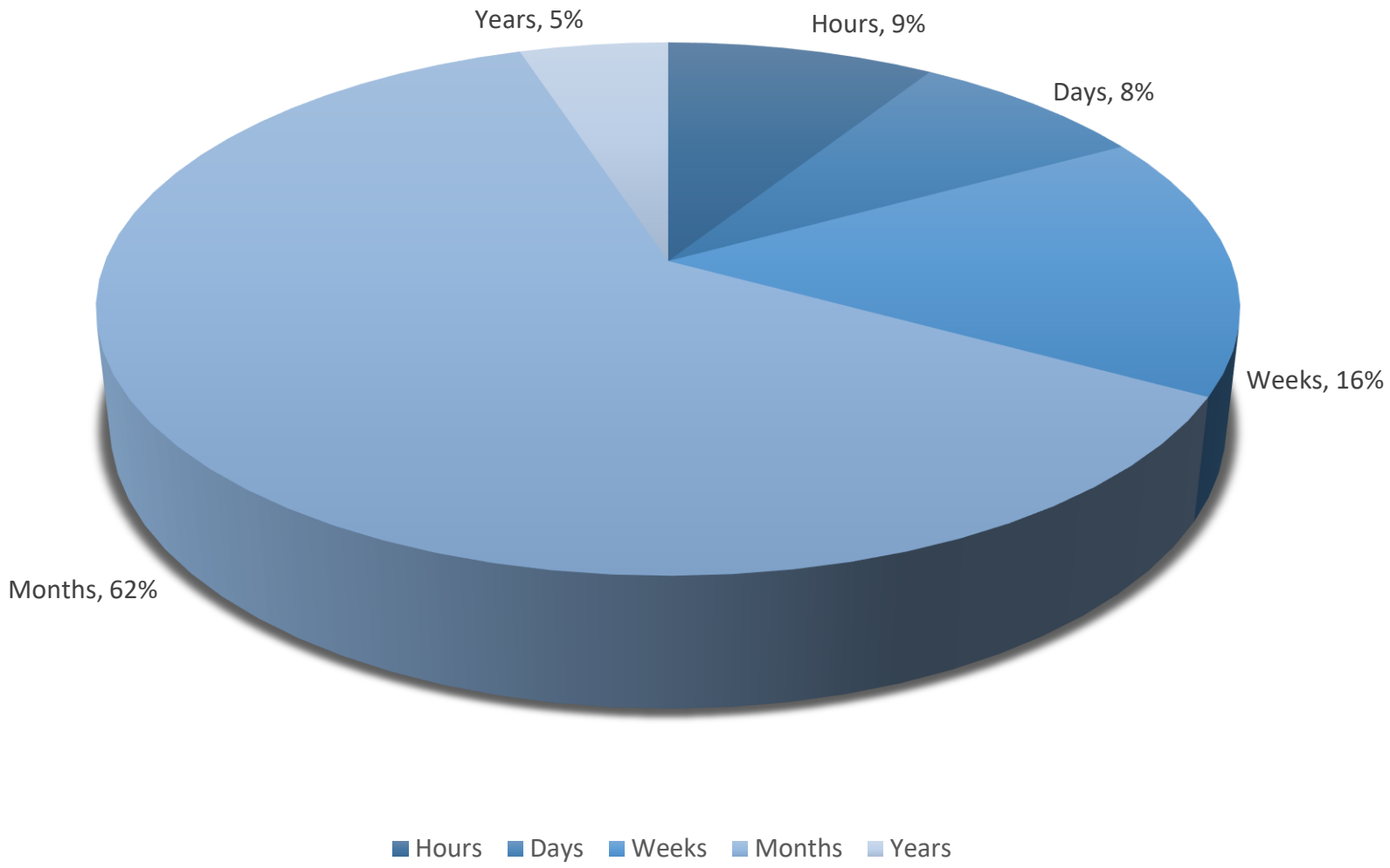
This could take weeks, or even months

Attack timeline



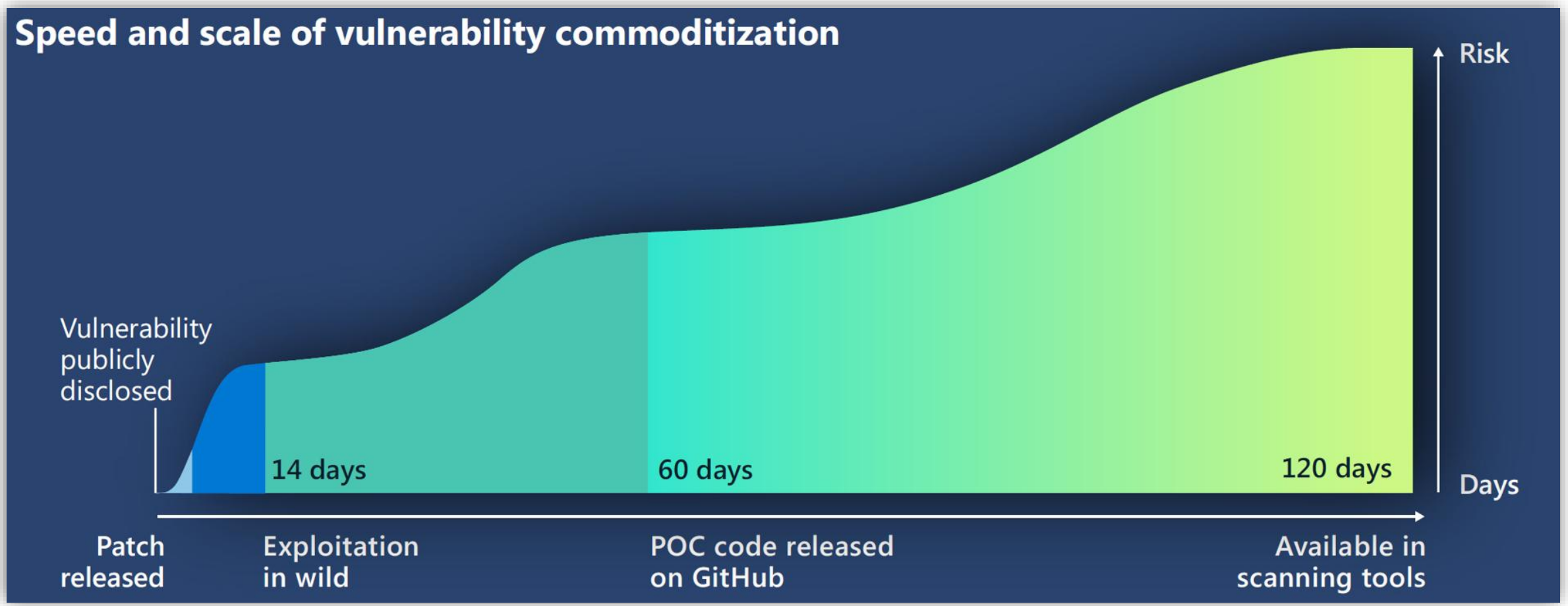
Sources: HP, Ponemon Institute, Verizon

Timeline of discovery for cyber attacks worldwide



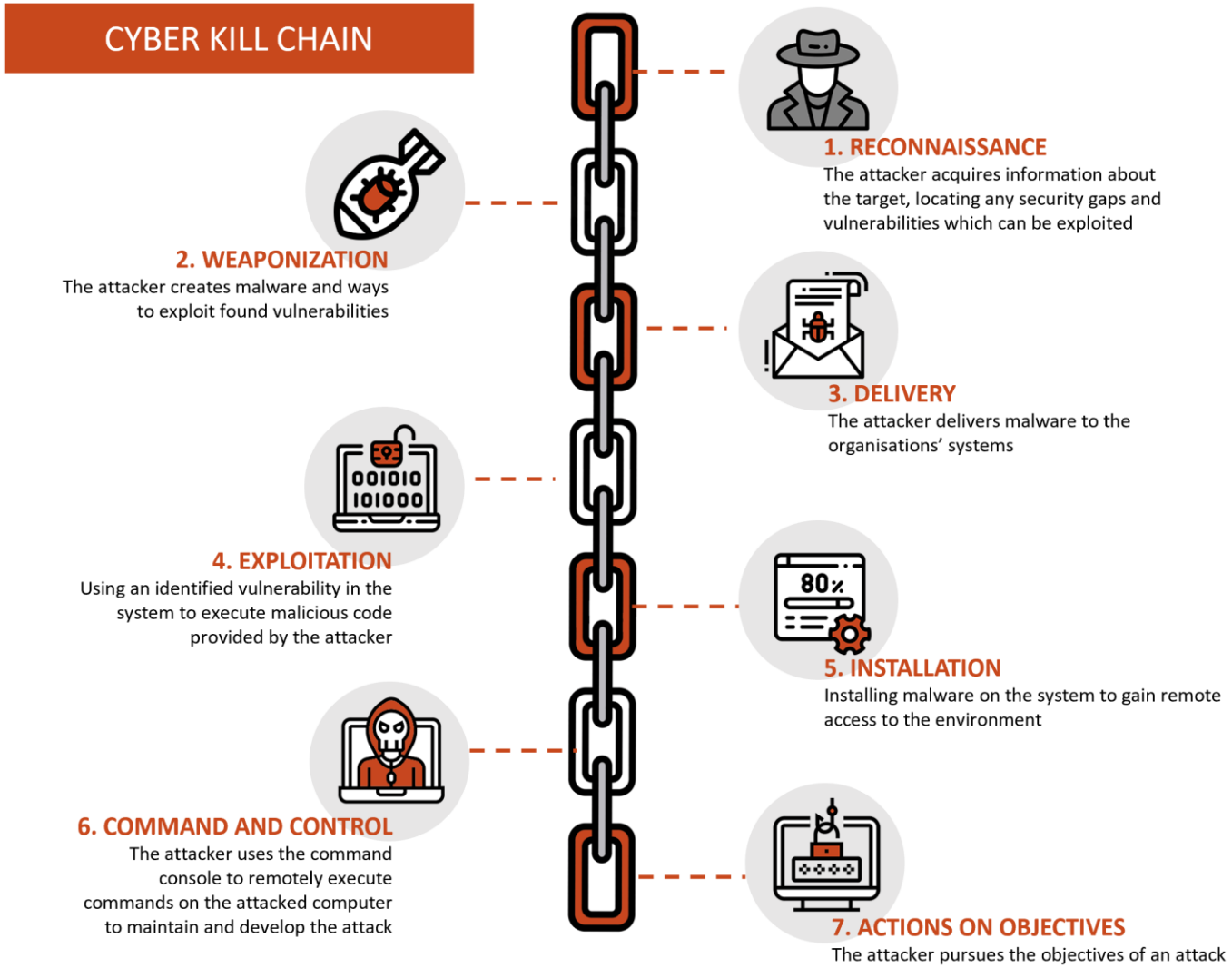
Source: Verizon

Rapid vulnerability Exploitation: Zero Days, today

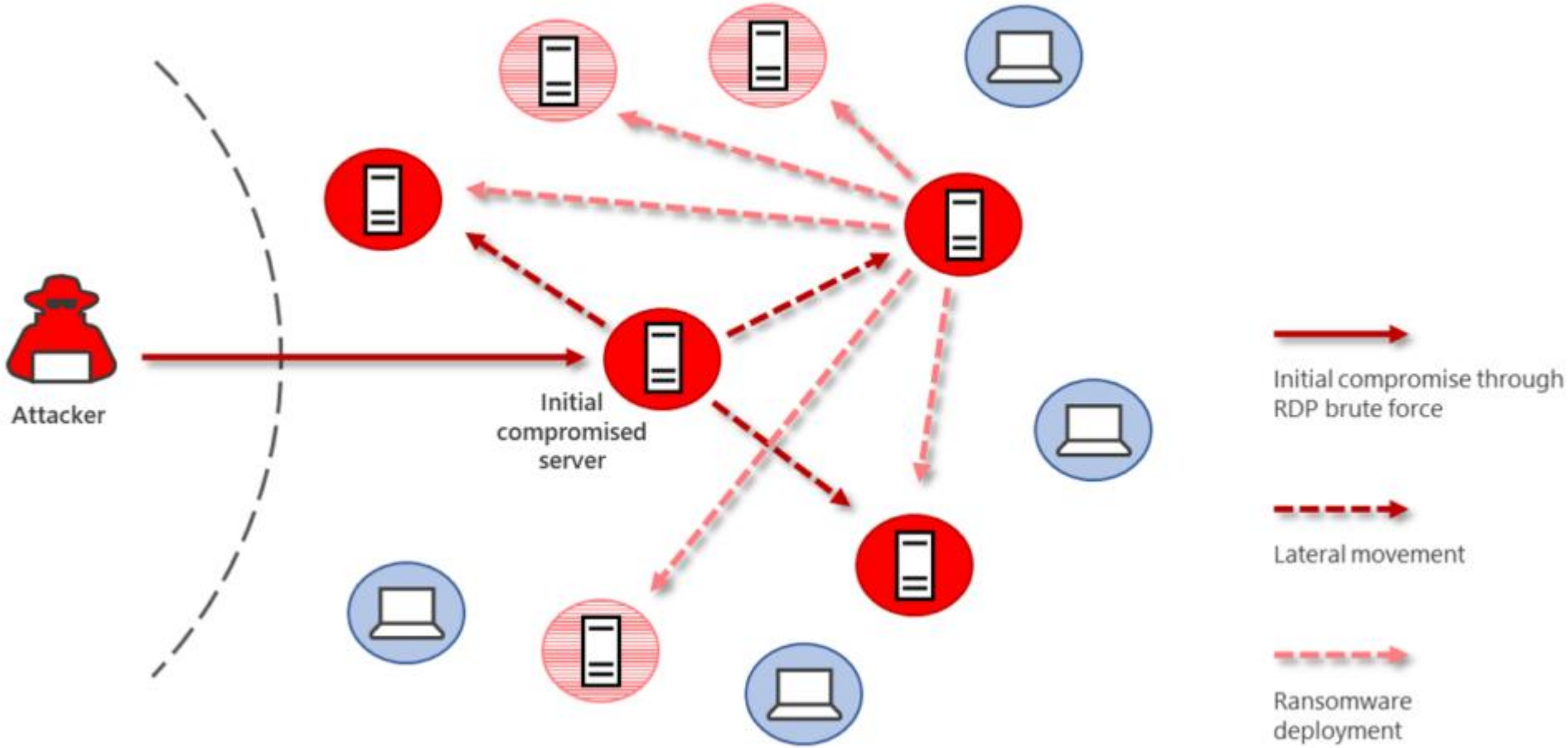


Source: Microsoft Digital Defense Report 2022

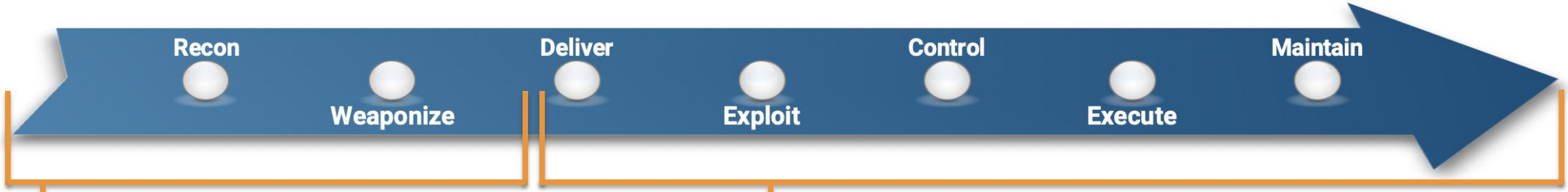
The Cyber Kill Chain



Example: RDP brute force + ransomware



MITRE ATT&CK framework



PRE-ATT&CK

- Priority Definition
  - Planning, Direction
- Target Selection
- Information Gathering
  - Technical, People, Organizational
- Weakness Identification
  - Technical, People, Organizational
- Adversary OpSec
- Establish & Maintain Infrastructure
- Persona Development
- Build Capabilities
- Test Capabilities
- Stage Capabilities

ATT&CK for Enterprise

- Initial Access
- Execution
- Persistence
- Privilege Escalation
- Defense Evasion
- Credential Access
- Discovery
- Lateral Movement
- Collection
- Exfiltration
- Command and Control
- Impact

Source: https://attack.mitre.org/matrices/enterprise/

MITRE ATT&CK - TACTICS

MITRE ATT&CK - biases

- **Novelty Bias** - Techniques or actors that are new or interesting are reported, while techniques that are being used over and over are not
- **Visibility Bias** - Intel report publishers have visibility biases that are based on how they gather data, resulting in visibility for some techniques and not others; additionally, techniques are also viewed differently during incidents and afterward
- **Producer Bias** - Reports published by some organizations may not reflect the broader industry or world as a whole
- **Victim Bias** - Some victim organizations are more likely to report, or to be reported on, than others
- **Availability Bias** - Report authors often include techniques that quickly come to mind in their reports

Hacking vs. Ethical Hacking

"With great power comes great responsibility"

(Quote from **Voltaire**. Or from **Spider-Man**. Your pick. 🙃)

- Hacking refers to exploiting system vulnerabilities and compromising security controls to gain unauthorized or inappropriate access to the system resources
  - It involves modifying system or application features to achieve a goal outside of the creator's original purpose

- Ethical Hacking involves the use of hacking tools, tricks, and techniques to identify vulnerabilities so as to ensure system security
  - It focuses on simulating techniques used by attackers to verify the existence of exploitable vulnerabilities in the system security

Hacking Phases (simplified)

Footprinting & Reconnaissance → Scanning & Enumeration → Gaining Access (Hacking) → Maintaining Access → Clearing Tracks

Security Assessments, Audits & Pentests

Security Assessments

- Every organization uses different types of security assessments to validate the level of security on its network and system resources

Security Assessments	Security Audits	Penetration Testing	Vulnerability Scans

Comparison (1/2)

Aspect	Security Assessment	Security Audit	Pentest	Vulnerability Scan
Objective	To identify security risks and improve security measures	To check if security measures comply with certain standards	To simulate cyberattacks and identify exploitable vulnerabilities	To automatically detect and list known vulnerabilities
Scope	Broad, covering policies, practices, and technologies	Narrow, focused on compliance with specific criteria	Targeted, often focused on particular systems or applications	Broad, scanning for known vulnerabilities in software or networks
Methodology	Holistic review of security posture	Adherence to formal standards and policies	Real-world attack scenarios	Automated tools scanning for known vulnerabilities
Outcome	Strategic recommendations for improvement	Report on compliance status and deviations	Detailed report on found vulnerabilities and how they were exploited	List of vulnerabilities with risk ratings
Frequency	Regularly, to keep up with changing threats	Periodically, based on regulatory or policy requirements	As needed, often annually or after major changes	Regularly, to identify new vulnerabilities as they become known

Comparison (2/2)

Aspect	Security Assessment	Security Audit	Pentest	Vulnerability Scan
Performed by	Security experts, often internal or external	Auditors specializing in security standards	Ethical hackers with specialized skills	Security personnel or automated systems
Follow-up	Implementation of improvements	Corrective actions for non-compliance	Remediation of exploited vulnerabilities	Patching or mitigating detected vulnerabilities
Focus	Overall security strategy	Compliance with specific standards	Depth of defense against an active attacker	Identification of known security weaknesses
Deliverables	Assessment report with strategic insights	Audit report with compliance findings	Penetration test report with exploitation details	Vulnerability scan report with a list of detected weaknesses
Regulatory Requirement	Not usually required by law but best practice	Often required by regulations or industry standards	Not typically mandated by law but critical for security	Not typically required by law but important for maintenance of security posture

What Should be Tested? (1/2)

Network Security	System Security	Application Security	Data Security	Identity and Access Management (IAM)
• Network topology analysis • Firewall configurations and ruleset reviews • Intrusion Detection/Prevention Systems (IDS/IPS) effectiveness • Network segmentation and access controls • Wireless network security • Virtual Private Network (VPN) gateways • Network protocols and services • External and internal network penetration testing	• Operating system hardening and configurations • Patch management processes • Anti-malware tools and controls • System access controls • Encryption for data at rest and in transit • Endpoint protection and response • File integrity monitoring	• Application code reviews • Web application penetration testing • Software composition analysis • Secure coding practices • Application patching processes • Input validation and output encoding checks • Authentication and authorization mechanisms • Session management security • Third-party services and APIs security	• Data classification and handling • Data encryption methods • Data loss prevention (DLP) strategies • Backup and restoration processes • Data privacy compliance • Database security configurations	• User account lifecycle management • Privileged access management • Multi-factor authentication (MFA) implementation • Identity federation and single sign-on (SSO) • Role-based access control (RBAC) policies

What Should be Tested? (2/2)

Physical Security	Operational Security	Compliance and Governance	Vendor and Third-Party Risk	Cloud and Virtualization Security
• Physical access controls to facilities • Surveillance and monitoring systems • Environmental controls (e.g., fire suppression, HVAC) • Protection against physical tampering	• Incident response planning and procedures • Business continuity and disaster recovery planning • Security information and event management (SIEM) implementation • Change management processes • Employee security awareness training	• Adherence to relevant industry standards and regulations (e.g., GDPR, HIPAA, PCI-DSS) • Policy review and enforcement • Risk assessment methodologies • Security audit logs and trail reviews	• Vendor security policies and controls • Third-party service provider assessments • Supply chain risk management	• Cloud configuration and security posture • Container and orchestration security • Cloud access security brokers (CASB) • Virtualization platform security

Just make sure you pick the right tool for the job ☺

Security Assessments

Security Audits

Penetration Testing

Vulnerability Scans

What Makes a Good Penetration Test?

- Establishing the parameters for the penetration test such as objectives, limitations and the justification of procedures
- Hiring skilled and experienced professionals to perform the test
- Choosing a suitable set of tests that balance cost and benefits
- Following a methodology with proper planning and documentation
- Documenting the result and making it comprehensible for the client
- Stating the potential risks and findings clearly in the final report

Types of Penetration Testing

External Testing

- External testing involves analysis of publicly available information, a network enumeration phase and the behavior of the security devices analyzed

Internal Testing

- Internal testing involves testing computers and devices within the company
- Black-hat testing / zero-knowledge testing
- Gray-hat testing / partial-knowledge testing
- White-hat testing / full-knowledge testing
- Announced testing
- Unannounced testing

Common Penetration Testing Techniques

- Passive Research
- Open Source Monitoring
- Network Mapping and OS Fingerprinting
- Spoofing
- Network Sniffing
- Trojan Attacks
- Brute-Force Attacks
- Vulnerability Scanning
- Scenario Analysis

We can’t test what we don’t understand

- **Business Logic** – Finite State Machines
  - Automated scanners are dumb
  - No idea of business state or state transitions
  - No clue about horizontal or vertical authorization / roles
  - No clue about business context
- We test applications for security issues without knowing the business process behind them
- **We can't “break” logic** (in a meaningful way) we don’t understand
- Running a $30,000 scanning tool against your mission critical application?
  - Will this find flaws in your business logic or state machine?
- We need **human intelligence** & **verification**

Pentesting Methodologies

Penetration Testing Execution Standard (PTES)

http://www.pentest-standard.org/

- Quite old, a second version is “in the works”

- Pre-engagement Interactions
- Intelligence Gathering
- Threat Modeling
- Vulnerability Analysis
- Exploitation
- Post Exploitation
- Reporting

PTES – Pre-engagement interactions (1)

http://www.pentest-standard.org/index.php/Pre-engagement

- Pre-engagement setup of a penetration test
- A penetration test should not be confrontational
  - Not an activity to see if the tester can “hack” you
- Purpose: identifying business risk associated with the attacks
- Required to have these at the end of the interactions:
  - A clear scope (usually after the contract/NDA has been signed)
    - Number of IPs/servers/applications, etc.
  - Rules of engagement
    - Compliance requirements, timeframes, existing limitations, in-place security
    - What happens when a vulnerability is identified?
    - These rules vary based on pentest type (network, web, wireless, physical, SE, etc.)
    - Locations, evidence handling, status meetings, time of day to test, permissions
  - Costs

PTES – Pre-engagement interactions (2)

- Beware of:
  - Scope creep (customer asking for more work, within the same scope)
  - Start & end dates
  - IP ranges & domains
  - Dealing with 3rd parties (Cloud Services, ISP, MSSPs, hosting location)
  - Acceptable SE pretexts
  - DoS testing
  - Payment terms (net 30, half upfront, recurring, etc.)
  - Goals (primary, secondary, business analysis)
  - Lines of communication
  - Emergency contact information
  - Incident reporting

PTES – Pre-engagement interactions (3)

- During and after a pentest, it's generally a good idea to also test the customer's ability to detect and respond to:
  - Information gathering
  - Footprinting
  - Scanning and vulnerability analysis
  - Infiltration (attacks)
  - Data aggregation and exfiltration

PTES – Intelligence Gathering (1)

http://www.pentest-standard.org/index.php/Intelligence_Gathering

- **Level 1 Information Gathering**
  - (think: Compliance Driven) Mainly a click-button information gathering process. This level of information can be obtained almost entirely by automated tools. Bare minimum to say you did IG for a PT.
  - Acme Corporation is required to be compliant with PCI / FISMA / HIPAA. A Level 1 information gathering effort should be appropriate to meet the compliance requirement.
- **Level 2 Information Gathering**
  - (think: Best Practice) This level can be created using automated tools from level 1 and some manual analysis. A good understanding of the business, including information such as physical location, business relationships, org chart, etc.
  - Widgets Inc is required to be in compliance with PCI, but is interested in their long term security strategy, and is acquiring several smaller widget manufacturers. A Level 2 information gathering effort should be appropriate to meet their needs.
- **Level 3 Information Gathering**
  - (think: State Sponsored) More advanced pentest, Redteam, full-scope. All the info from level 1 and level 2 along with a lot of manual analysis. Think cultivating relationships on SocNet, heavy analysis, deep understanding of business relationships, most likely a large number of hours to accomplish the gathering and correlation.
  - An Army Red Team is tasked to analyze and attack a segment of the Army's network in a foreign country to find weaknesses that could be exploited by a foreign national. A level 3 information gathering effort would be appropria this case.

PTES – Intelligence Gathering (2)

- What it is
  - Performing reconnaissance against a target to gather as much information as possible
  - Info will be utilized when penetrating the target during the vulnerability assessment and exploitation phases
  - Open source intelligence (OSINT) is a form of intelligence collection management that involves finding, selecting, and acquiring information from publicly available sources and analyzing it to produce actionable intelligence
- Why do it
  - In order to determine various entry points into an organization (physical, electronic, and/or human)
  - Many companies fail to take into account what information about themselves they place in public and how this information can be used by a determined attacker
  - On top of that many employees fail to take into account what information they place about themselves in public and how that information can be used to attack them or their employer
- What is it not
  - OSINT may not be accurate or timely
  - The information sources may be deliberately/accidentally manipulated to reflect erroneous data, information may become obsolete as time passes, or simply be incomplete
  - It does not encompass dumpster-diving or any methods of retrieving company information off of physical items found on-premises

https://en.wikipedia.org/wiki/Open-source_intelligence

PTES – Intelligence Gathering (3)

- Open Source Intelligence (OSINT) takes three forms: Passive, Semi-passive, and Active
  - Passive Information Gathering: Passive Information Gathering is generally only useful if there is a very clear requirement that the information gathering activities never be detected by the target
    - Technically difficult to perform as we are never sending any traffic to the target organization neither from one of our hosts or "anonymous" hosts or services across the Internet
    - We can only use and gather archived or stored information
    - This information can be out of date or incorrect as we are limited to results gathered from a third party
  - Semi-passive Information Gathering: The goal for semi-passive information gathering is to profile the target with methods that would appear like normal Internet traffic and behavior
    - We query only the published name servers for information, we aren't performing in-depth reverse lookups or brute force DNS requests, we aren't searching for "unpublished" servers or directories
    - We aren't running network level port scans or crawlers and we are only looking at metadata in published documents and files; not actively seeking hidden content
    - The key here is not to draw attention to our activities
    - Post mortem the target may be able to go back and discover the reconnaissance activities but they shouldn't be able to attribute the activity back to anyone
  - Active Information Gathering: Active information gathering should be detected by the target and suspicious or malicious behavior
    - During this stage we are actively mapping network infrastructure (think full port scans nmap –p1-65535), actively enumerating and/or vulnerability scanning the open services, we are actively searching for unpublished directories, files, and servers
    - Most of this activity falls into your typically "reconnaissance" or "scanning" activities for your standard pentest

PTES – Intelligence Gathering (4)

- **Corporate**
  - Physical (Locations, Pervasiveness, Relationships)
  - Electronic (document metadata, marketing communications)
  - Infrastructure assets (network blocks, email addresses, technologies used, remote access, application usage, defense technologies)
  - Financial
- **Individual**
  - Employee (history, social profile, Internet presence, mobile footprint)
- **Footprinting**
  - External (IP ranges, WHOIS/BGP, port scanning, banner grabbing, SNMP sweeps, zone transfers, DNS discovery, DNS bruteforce, web app discovery, vhost detection, versions, patch levels, lockout thresholds, network/host/application protection, etc.

PTES – Threat Modeling (1)

http://www.pentest-standard.org/index.php/Threat_Modeling

- The part of the penetration test process that focuses on:
  - Business assets
  - Business procedures
  - Attackers (threat communities or agents)
  - Attacker capabilities

- High level threat modeling process
  - Gather relevant documentation
  - Identify and categorize primary and secondary assets
  - Identify and categorize threats and threat communities
  - Map threat communities against primary and secondary assets

PTES – Threat Modeling (2)

Business Asset Analysis

- Organizational Data (Policies, Plans, Procedures)
- Product Information (trade secrets, R&D data)
- Marketing Information (plans, roadmaps, etc.)
- Financial Information (bank, credit, equity accounts, etc.)
- Technical Information
  - Infrastructure Design Information
  - System Configuration Information
  - User Account Credentials
  - Privileged User Account Credentials
- Employee Data
- Customer Data
- Human Assets

PTES – Threat Modeling (3)

Business Process Analysis

- Technical infrastructure supporting process
  - Networks, processing powers, infrastructure entry and access points
- Information assets supporting process
  - Service delivery, decision making, legal, marketing, financial, etc.
- Human assets supporting process
- 3rd party integration and/or usage of/by process
  - Human assets, vendors/contractors, SaaS providers

PTES – Threat Modeling (4)

Threat Agents/Community Analysis

Internal	External
Employees	Business Partners
Management (executive, middle)	Competitors
Administrators (network, system, server)	Contractors
Developers	Suppliers
Engineers	Nation States
Technicians	Organized Crime
Contractors (with their external users)	Hacktivists
General user community	Script Kiddies (recreational/random hacking)
Remote Support	

PTES – Threat Modeling (5)

Threat Capability Analysis

- Analysis of tools in use
- Availability to relevant exploits/payloads
- Communication mechanisms
- Accessibility

Motivation Modeling

- Profit (direct or indirect)
- Hacktivism
- Direct grudge
- Fun / Reputation
- Further access to partner/connected systems

PTES – Vulnerability Analysis (1)

http://www.pentest-standard.org/index.php/Vulnerability_Analysis

- Process of discovering flaws in systems and applications which can be leveraged by an attacker
  - Flaws can range anywhere from host and service misconfigurations to insecure application design
  - Ideal to have the scope and threat analysis done right before you start

- Usually one of the most complex sides of the penetration test
  - Should be tailored to meet the depth and breadth requirements needed to reach your goals

PTES – Vulnerability Analysis (2)

Active analysis

- Automated
- Network/General Vulnerability Scanners (port/service based)
- Banner Grabbing
- Web Application Scanners
- Directory Listing / Brute Forcing
- Web Server Version/Vulnerability Identification
- Manual Direct Connections
- Obfuscated (e.g. IDS Evasion)

Passive analysis

- Metadata Analysis
- Traffic Monitoring

PTES – Vulnerability Analysis (3)

Validation

- Correlation between tools
- Manual Testing / Protocol Specific Testing
  - VPN, DNS, web, email, etc.

Attack Avenues

- Creation of attack trees
- Isolated lab testing
- Visual confirmation

PTES – Vulnerability Analysis (4)

Research

- Public research
  - Vulnerability databases
  - Vendor advisories
- Exploit databases and framework modules
- Common/default passwords
- Hardening guides & common misconfigurations
- Private research
  - Setting up a replica configuration
  - Testing configurations
  - Fuzzing (fault injection)
- Identifying potential avenues/vectors
- Disassembly and code analysis

PTES – Exploitation (1)

http://www.pentest-standard.org/index.php/Exploitation

- The exploitation phase focuses solely on establishing access to a system or resource by bypassing security restrictions
  - If vulnerability analysis was performed properly, this phase should be well planned and a precision strike
- Countermeasures
  - Host & Network IPS, Security Guard, WAF, other preventative methods
  - Antivirus, Encoding, Packing, Encryption, Whitelist Bypass, Process Injection, Purely Memory Resident, Data Execution Prevention (DEP), Address Space Layout Randomization (ASLR), Human layer
- Evading IDS/IPS
- Tailored Exploits
  - Exploit customization
- Zero-Day Angle
  - Fuzzing, source code analysis, traffic analysis, physical access, proximity access (WiFi)

PTES – Exploitation (2)

Example Avenues of Attack

- Known Vulnerabilities
- Server/Service Attacks
- Web Application Attacks
- Social Engineering
- Physical Attack Avenues
- Memory Based Exploits (buffer/heap overflows, memory corruptions)
- Man-in-the-Middle (MitM)
- VLAN hopping
- USB/Flash drive deployment
- Reverse Engineering
- Zero-Day
- Attacking the User
- Encryption Cracking
- Traffic Analysis
- Firewire
- Routing Protocols
- Phishing
- Employee Impersonation
- etc.

PTES – Post Exploitation (1)

http://www.pentest-standard.org/index.php/Post_Exploitation

- The purpose of this phase is to determine the value of the compromised machines or components of the infrastructure, as well as maintain control (via a backdoor) for later use
  - Value is determined by the sensitivity of the data stored on it, as well as its usefulness in compromising the rest of the network
- Rules of Engagement
  - Protect the client
  - Protect yourself

PTES – Post Exploitation (2)

Protect the client

- No modification of scope, unless previously agreed upon
- All modifications, including config changes, must be documented
- A list of actions taken against compromised systems must be kept
- Avoid using any tool/backdoor that could cause downtime to remove
- All collected data must be securely stored
- Any information included in the report must be sanitized/masked
- All data gathered will be destroyed once customer accepts final report
- If evidence of a prior compromise is found, all logs with actions and times recorded during the assessment will be provided to the client
- No logs should be removed, cleared or modified unless specifically authorized to do so by the client in the contract/SOW

PTES – Post Exploitation (3)

Protect yourself

- Ensure the contract/SOW states that the actions taken on the system being taken are on behalf and in representation of the client
- Obtain a copy of the security policy that govern the use of company systems and infrastructure (Acceptable Use Policies)
- Confirm regulations and laws that govern the data managed and used by the client, and any restrictions imposed on that data
- Use full drive encryption for the systems and removable media that will receive and store client data
- Discuss and establish with the client the procedures to follow in case a compromise from a 3rd party is found during the penetration test

PTES – Post Exploitation (4)

Infrastructure Analysis

- Network Configuration
  - Interfaces, routing, DNS servers, cached DNS entries, proxy servers, ARP, DHCP
- Network Services
  - Listening services, VPN connections, Directory Services, neighbors (CDP – Cisco Discovery Protocol, LLDP – Link Layer Discovery Protocol, NetBIOS, multicast DNS)
- Pillaging (obtaining information from target hosts)
  - Installed programs, Startup items, Installed Services, Security Services, File/Printer shares, DB servers, Directory Servers, Name Servers, Deployment Services, Certificate Authorities, Source Code management servers, DHCP servers, Virtualization, Messaging, Monitoring and Management, Backup Systems, Networking Services (e.g. RADIUS), Password Policy
- Sensitive data
  - Keylogging, screen capture, network traffic capture
- User information
  - On system (history, encryption keys, individual app history, removable media, shares), web browsers (bookmarks, history, credentials), IM clients (chat logs), WiFi keys

PTES – Post Exploitation (5)

Data Exfiltration

- Mapping all possible exfiltration paths
- Testing exfiltration paths
- Measuring control strengths
- Avoiding DLP systems (Data Leak Prevention)

Persistence

- Installing a backdoor that can ideally survive reboots
- Creating alternate accounts

PTES – Post Exploitation (5)

Further penetration into infrastructure

- Pivoting (enumerate, scan, compromise other systems)
- From compromised system
  - Upload tools
  - Use local system tools
  - ARP scan
  - Ping sweep
  - Enumeration on DNS, Directory Services, WinRM, WMI, SMB, SNMP, etc.
  - Brute Force
  - Abuse of compromised credentials and keys (websites, DBs)
  - Execute remote exploits
- Through compromised system
  - Port forwarding, proxy, VPN, etc.

PTES – Post Exploitation (6)

Cleanup

- Remove all executable, scripts and temp files from compromised systems
  - If possible, use secure delete methods
- Return system settings and app parameters to original values
- Remove all backdoors and/or rootkits
- Remove any user accounts created for connecting back

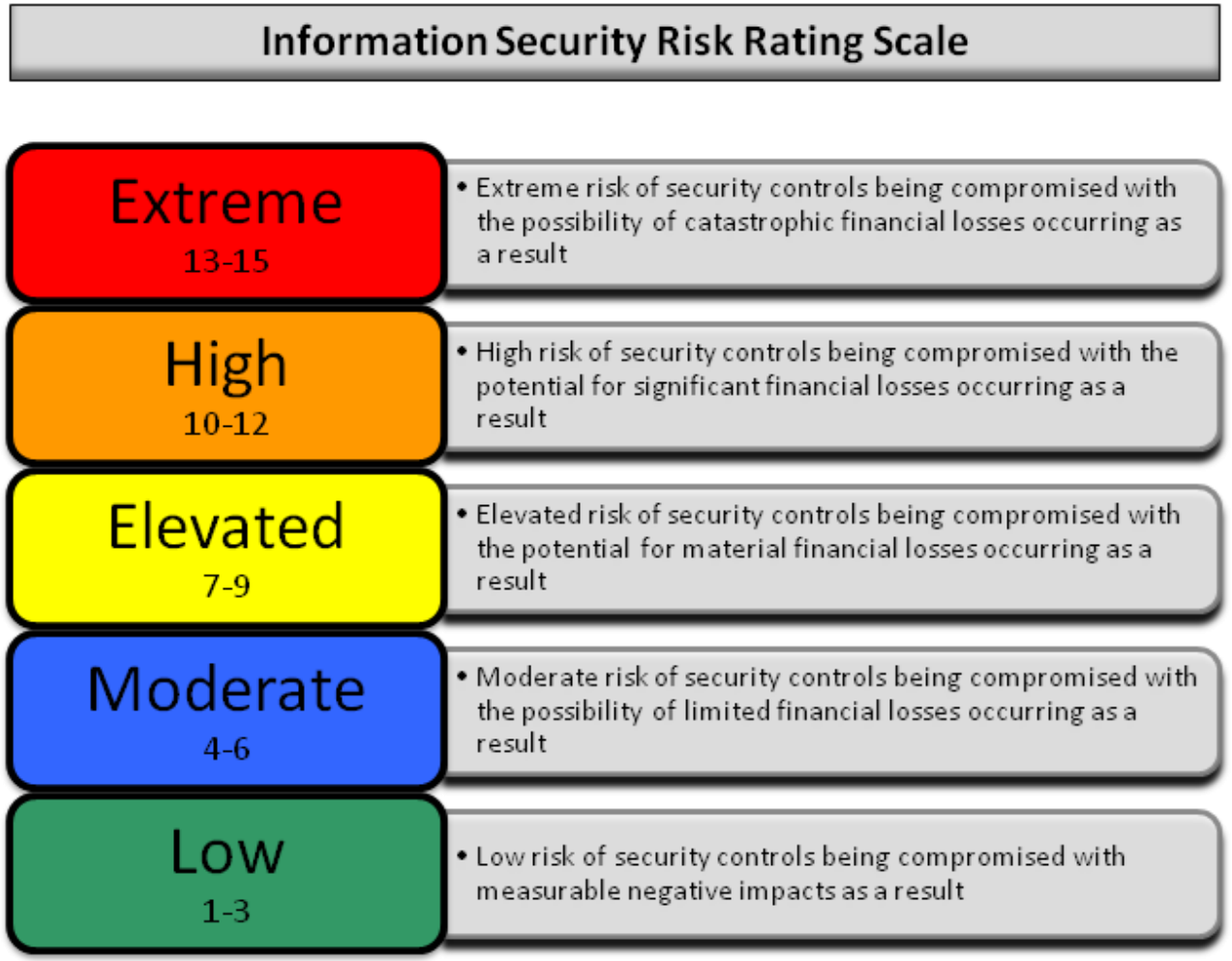
PTES – Reporting (1)

http://www.pentest-standard.org/index.php/Reporting

Suggested report structure

- Executive Summary (background, security posture, risk profile, general findings, recommendations, roadmap)
- Technical Report (intro, personnel involved, assets tested, objectives and scope, approach, threat/grading structure)
- Information Gathering
- Passive & Active Intelligence
- Corporate & Personnel Intelligence
- Vulnerability Assessment
- Exploitation/Vulnerability Confirmation
- Post Exploitation
- Risk/Exposure
- Conclusion

PTES – Reporting (2)



PTES – Technical Guidelines

http://www.pentest-standard.org/index.php/PTES_Technical_Guidelines

- A comprehensive set of resources and tools needed to perform an end-to end penetration test

PCI DSS - Overview

https://www.pcisecuritystandards.org/

- Payment Card Industry – Data Security Standards
  - A common set of industry tools and measurements to help ensure the safe handling of sensitive information
  - Provides an actionable framework for developing a robust account data security process - including preventing, detecting and reacting to security incidents.
  - Applies to any entity that stores, processes and/or transmits CHD (card holder data) – merchants, banks, processors, developers
- It is an industry regulation
  - The major Card Brands (Visa, MC, Discover, Amex) decided to create regulations which were initially agreed upon by all of them in 2004
  - PCI DSS v1 is dated Dec 2004, regulations took effect on June 30, 2005

PCI DSS - PCI Security Standards Council

- The PCI Security Standards Council came into existence in 2006
- The Council became responsible for the development, management, education and awareness of the PCI DSS
  - Each of the Card Brands (Visa, MC, Discover, Amex, JCB) have their own compliance programs in accordance with their own security risk management policies as well as their own definitions of the “levels” and their own penalizing/fining procedures for companies who have a breach

PCI DSS – Merchant Levels

- **Level 4**
  - Little credit card business
  - Some Card Brands do not have this level
  - Annual Compliance Validation
- **Level 3**
  - Less than a million credit card transactions
  - Some Card Brands do not have this level
  - Annual Self-Assessment
- **Level 2**
  - Millions (1+ to <6) credit card transactions
  - All Card Brands have this level
  - Must internally audit with a PCI certified Internal Security Assessor (ISA) using PCI DSS
- **Level 1**
  - Many millions (2.5+ to 6+) credit card transactions
  - All Card Brands have this level
  - Must audit either using a PCI certified external Qualified Security Assessor (QSA) OR Internal Audit with ISA certification using PCI DSS

PCI DSS - Goals

Goals	PCI DSS Requirements
Establish and maintain a secure network	**1.** Install firewalls and web filtering to protect cardholder data **2.** Change default or vendor-supplied device security configurations
Protect payment card and cardholder data	**3.** Protect cardholder data stored on company servers or networks **4.** Encrypt and protect cardholder data transmitted over open and public networks
Maintain a vulnerability management program	**5.** Use and keep up-to-date antivirus and malware software to protect cardholder data **6.** Develop and maintain secure systems and applications; use secure protocols everywhere
Implement strong identity & access control measures	**7.** Restrict access to cardholder data by need-to-know **8.** Identify and authenticate access to system components **9.** Restrict physical access to cardholder data
Regularly monitor and test networks and traffic	**10.** Track and monitor all access to network resources and cardholder data **11.** Regularly test security systems and processes
Maintain an information security policy	**12.** Maintain a policy that addresses information security for all personnel

Penetration Testing Framework

http://www.vulnerabilityassessment.co.uk/Penetration%20Test.html

- A very comprehensive, one-page reference for an end-to end penetration test, including tools, steps & examples
- Covers topics such as:
  - Network Footprinting
  - Discovery & Probing
  - Enumeration
  - Password Cracking
  - Vulnerability Assessment
  - Server specific tests
  - Bluetooth, Cisco, Citrix, VoIP, Wireless specific tests
  - Physical Security
  - Report template

NIST 800-115

- **Process and technical guidance** enabling organizations to:
  - Develop information security assessment policy, methodology, and individual roles and responsibilities related to the technical aspects of assessment
  - Accurately plan for a technical information security assessment by providing guidance on determining which systems to assess and the approach for assessment, addressing logistical considerations, developing an assessment plan, and ensuring legal and policy considerations are addressed
  - Safely and effectively execute a technical information security assessment using the presented methods and techniques, and respond to any incidents that may occur during the assessment
  - Appropriately handle technical data (collection, storage, transmission, and destruction) throughout the assessment process
  - Conduct analysis and reporting to translate technical findings into risk mitigation actions that will improve the organization's security posture.

NIST 800-115 – Recommendations

- Establish an information security assessment policy
- Implement a repeatable and well-documented assessment methodology
- Determine the objectives of each security assessment, and tailor the approach accordingly
- Analyze findings, and develop risk mitigation techniques to address weaknesses

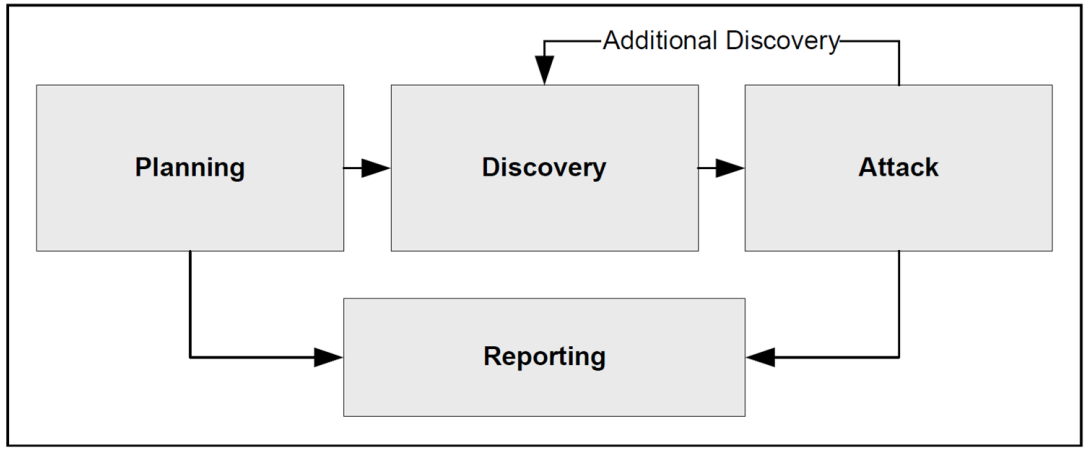


Figure 5-1. Four-Stage Penetration Testing Methodology

NIST 800-115 – Attack Phases

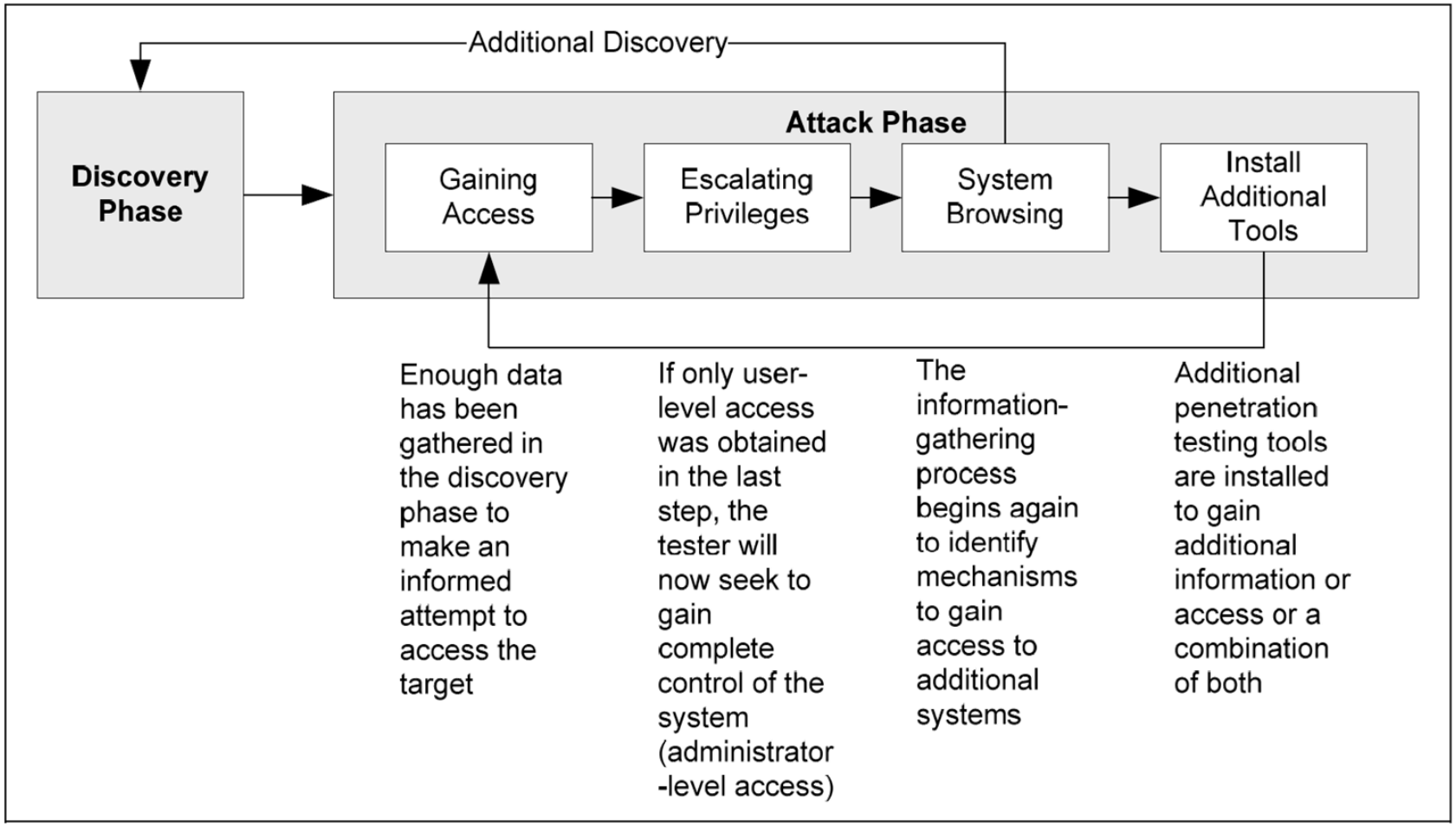
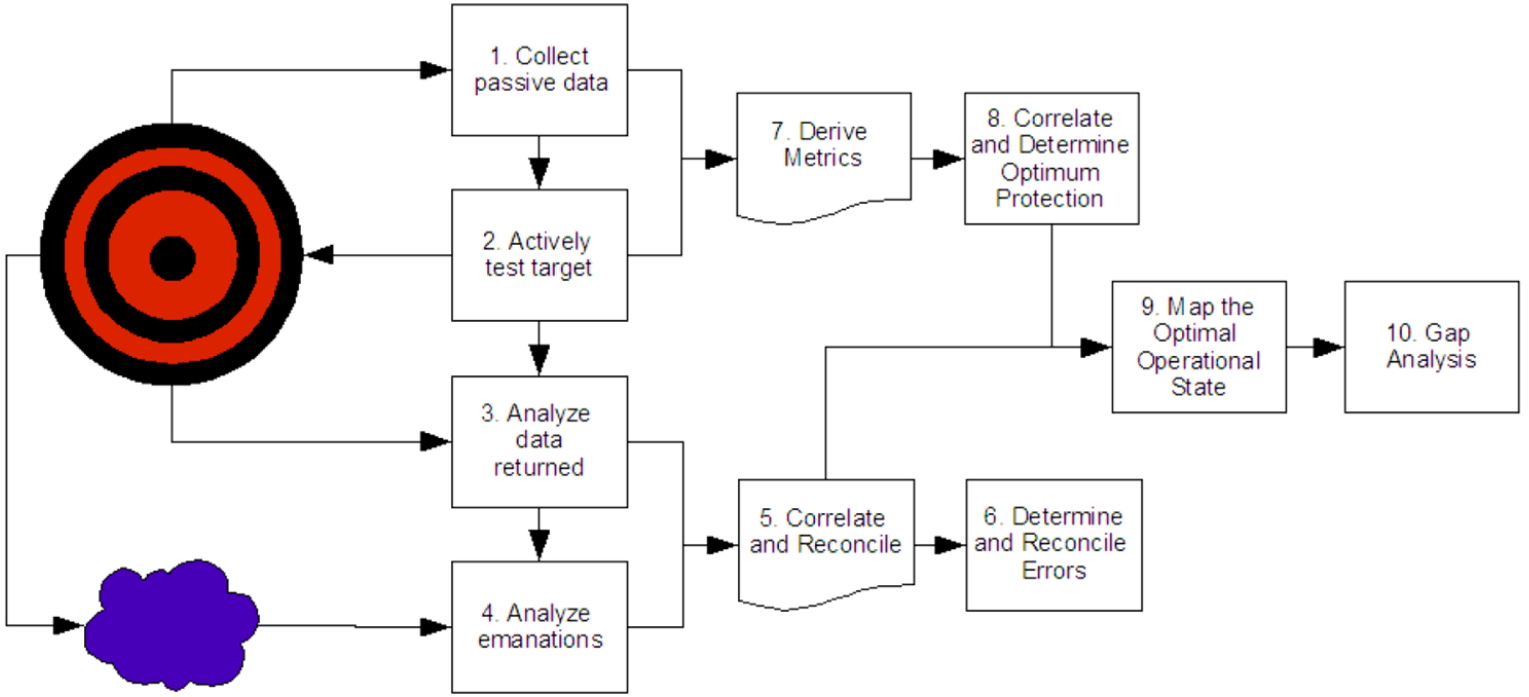


Figure 5-2. Attack Phase Steps with Loopback to Discovery Phase

OSSTMM 3.0

https://www.isecom.org/OSSTMM.3.pdf

Open Source Security Testing Methodology Manual (OSSTMM)



Combining the Trifecta and the 4 Point Process

Source: OSSTMM 3.0

MITRE ATT&CK framework

https://attack.mitre.org/

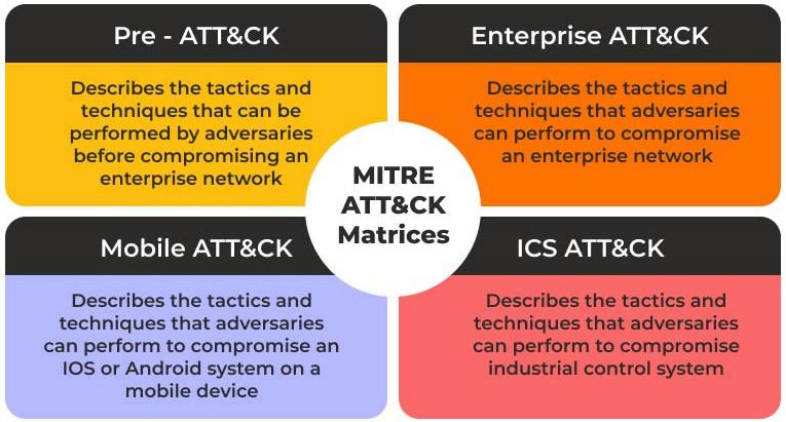
- MITRE's **Adversarial Tactics, Techniques & Common Knowledge** is a widely adopted framework and knowledge base that outlines and categorizes the tactics, techniques, and procedures (TTPs) used in cyberattacks

- Derived from real-world evidence of attacker's behaviors
- Used for:
  - Awareness
  - Threat Actor Analysis
  - Gap Analysis
  - Atomic Testing w/ Atomic Red Team

MITRE ATT&CK Framework

https://attack.mitre.org/

- Provides a structured taxonomy of real-world tactics and techniques used by threat actors
- Helps pentesters emulate sophisticated cyber threats
- Enhances red teaming capabilities
- Supports automated adversary emulation through MITRE CALDERA



Pentesting Tools

Kali Linux

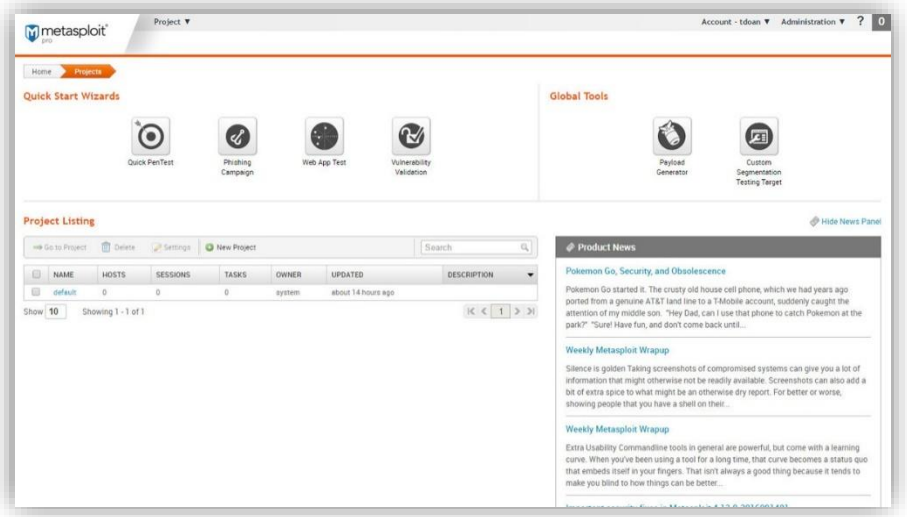
- Open source project, maintained by Offensive Security
  - They also maintain Exploit DB and Metasploit Unleashed
- A Debian-derived Linux distro
- Designed for digital forensics and penetration testing
- Preinstalled with over 300 programs, grouped in categories:
  - Information Gathering
  - Vulnerability Analysis
  - Wireless Attacks
  - Web Applications
  - Exploitation Tools
  - Forensics Tools
  - Stress Testing
  - Sniffing & Spoofing
  - Password Attacks
  - Maintaining Access
  - Reverse Engineering
  - Hardware Hacking
  - Reporting Tools

https://www.kali.org/tools/

KALI
BY OFFENSIVE SECURITY

Metasploit

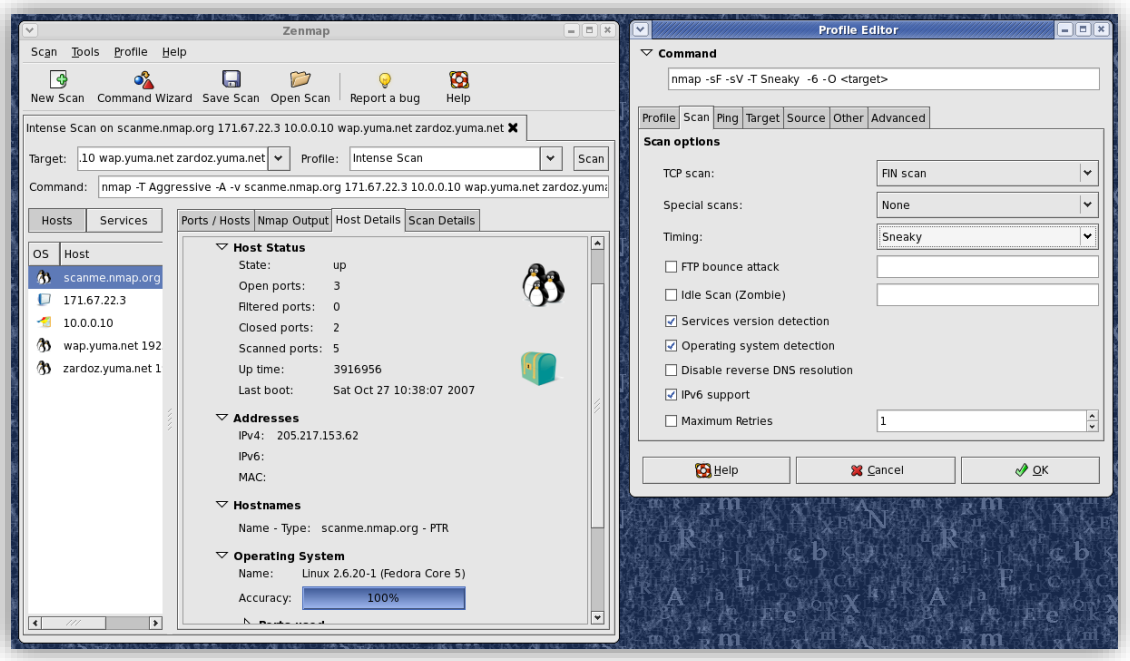
- World's most used offensive penetration testing software
  - Automate most steps of pentest
  - Create sophisticated attacks to test user weaknesses (site cloning, phishing campaigns, masking malicious files for USB drops, etc.)
  - Convenient tool, requiring little experience with the actual exploits
- A free training is offered by Offensive Security (Kali Linux)
  - https://www.offensive-security.com/metasploit-unleashed/



Nmap – Network Mapper

https://nmap.org/

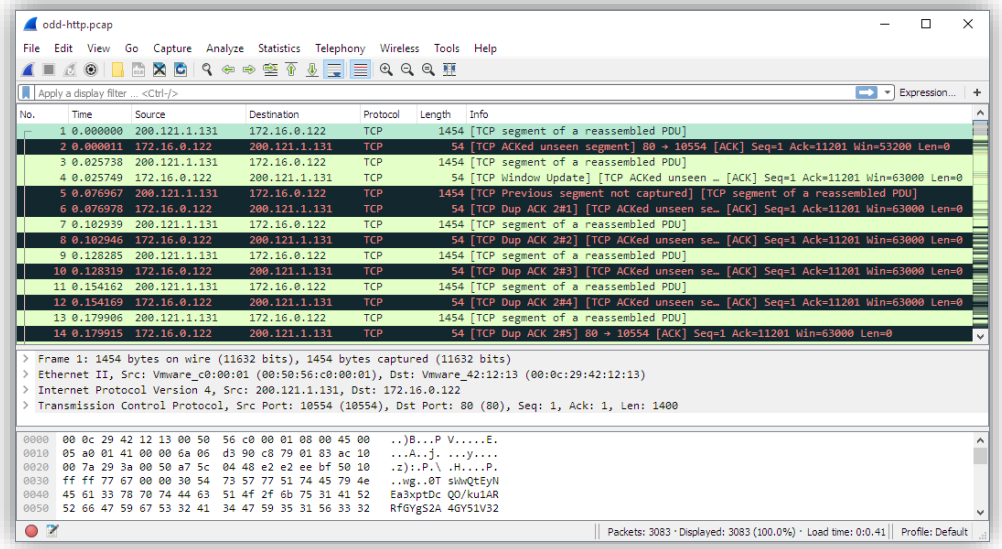
- Free and open source utility used for network discovery and security auditing
- Uses raw IP packets in novel ways to determine:
  - Hosts are available on the network
  - Services those hosts are offering
  - OS versions
  - Packet filters & firewalls in use
  - Known vulnerabilities
- Also has a GUI version (Zenmap)



Wireshark

https://www.wireshark.org/

- World's foremost network protocol analyzer
- Multi-platform: Windows, Linux, MacOS, Solaris, FreeBSD, NetBSD
- Lets you see what's happening on your network
  - The de facto standard across many enterprises
- Also works with AirPCap for 802.11 (WiFi) capture

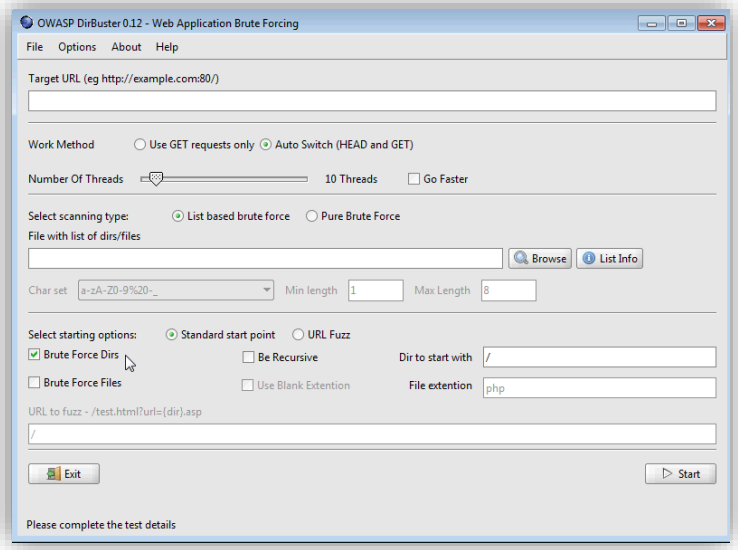


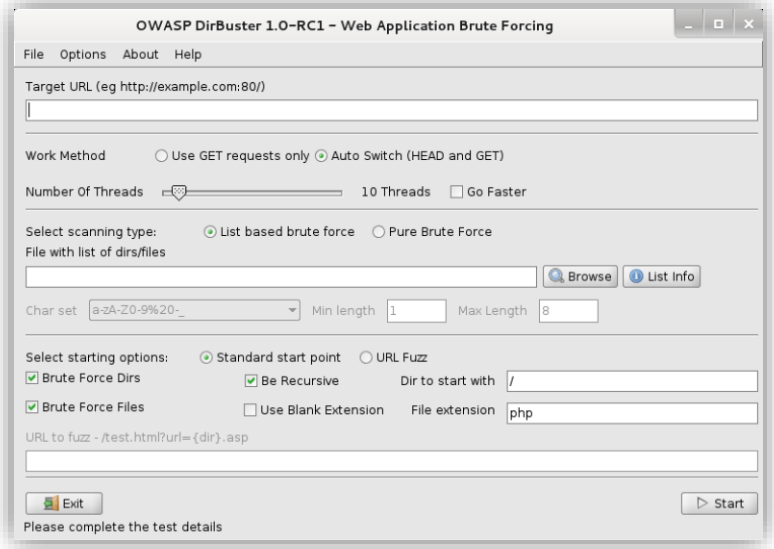
OWASP DirBuster

https://www.kali.org/tools/dirbuster/

A multi threaded java application designed to brute force directories and files names on web/application servers

- Forked by the OWASP ZAP and included as an add-on
- Included in Kali Linux



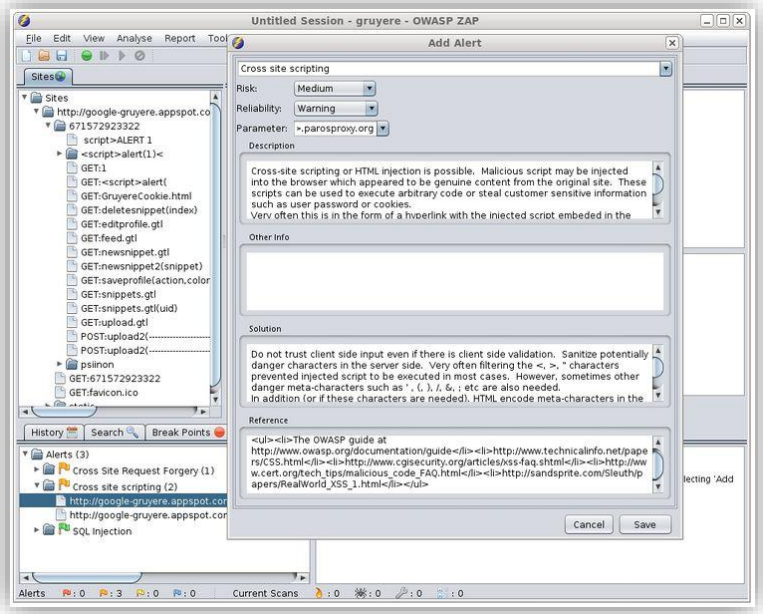


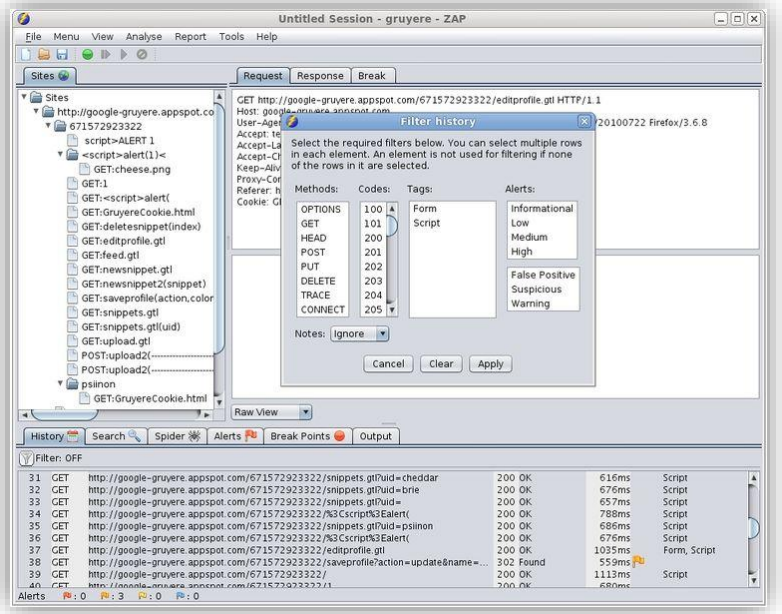
OWASP Zed Attack Proxy (ZAP)

https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project

https://github.com/zaproxy/

- One of the world's most popular free security tools
- Helps you find security vulnerabilities in web applications
- Used for manual security testing by experienced pentesters

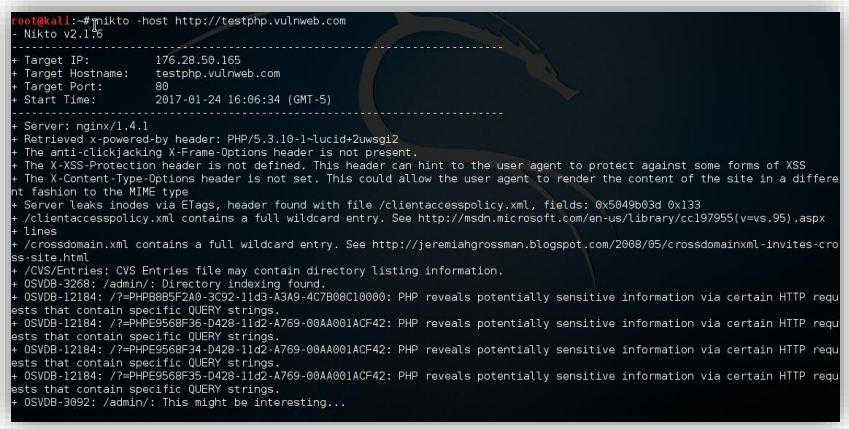




Nikto

https://github.com/sullo/nikto

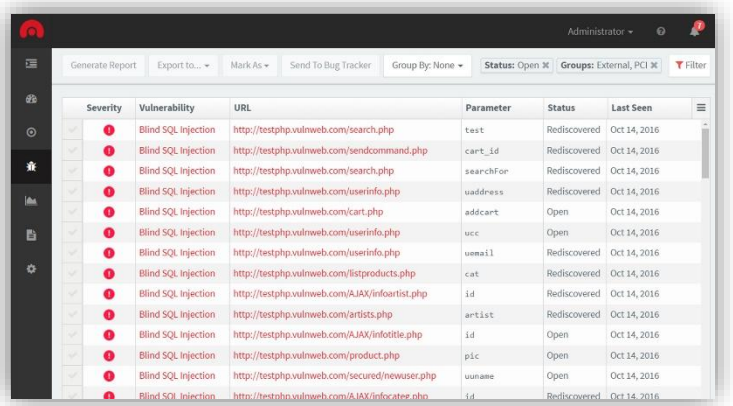
- Open source web scanner which performs comprehensive tests against web servers for multiple items, such as:
  - Over 6400 potentially dangerous files/CGIs
  - Outdated versions on over 1200 servers
  - Version specific problems on over 270 servers
- Designed as a stealthy tool
  - Will test a web server in the quickest time possible
  - Supports LibWhisker's anti-IDS methods
  - Can perform logging directly to Metasploit



Acunetix

https://www.acunetix.com

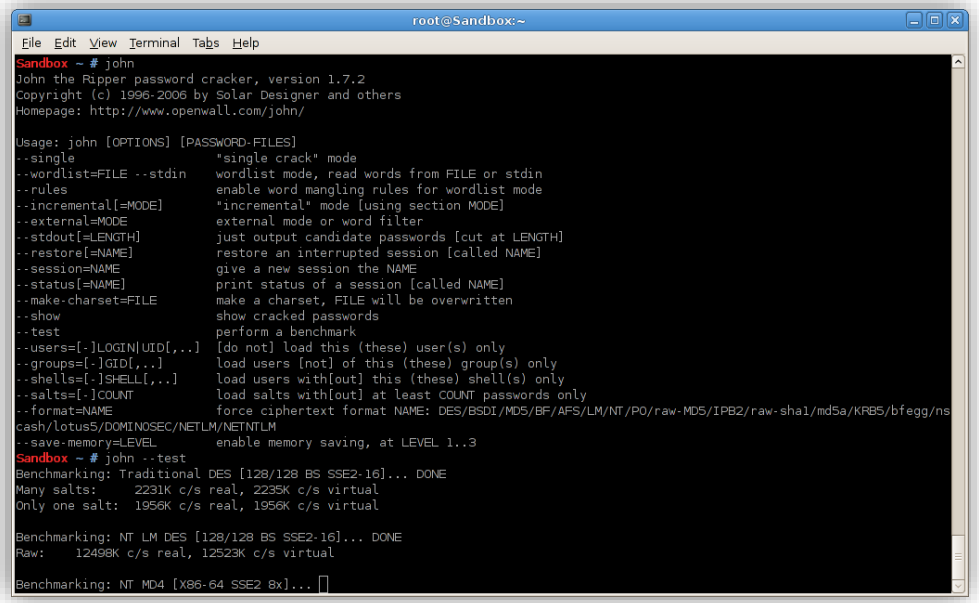
- Web & Network Vulnerability Scanner
  - A good number of vulnerabilities
  - A decently low false positive rate
- They've switched to a web-based model in 2016
- They also made a bunch of manual tools available for free
  - http://www.acunetix.com/vulnerability-scanner/free-manual-pen-testing-tools/



John The Ripper

http://www.openwall.com/john/

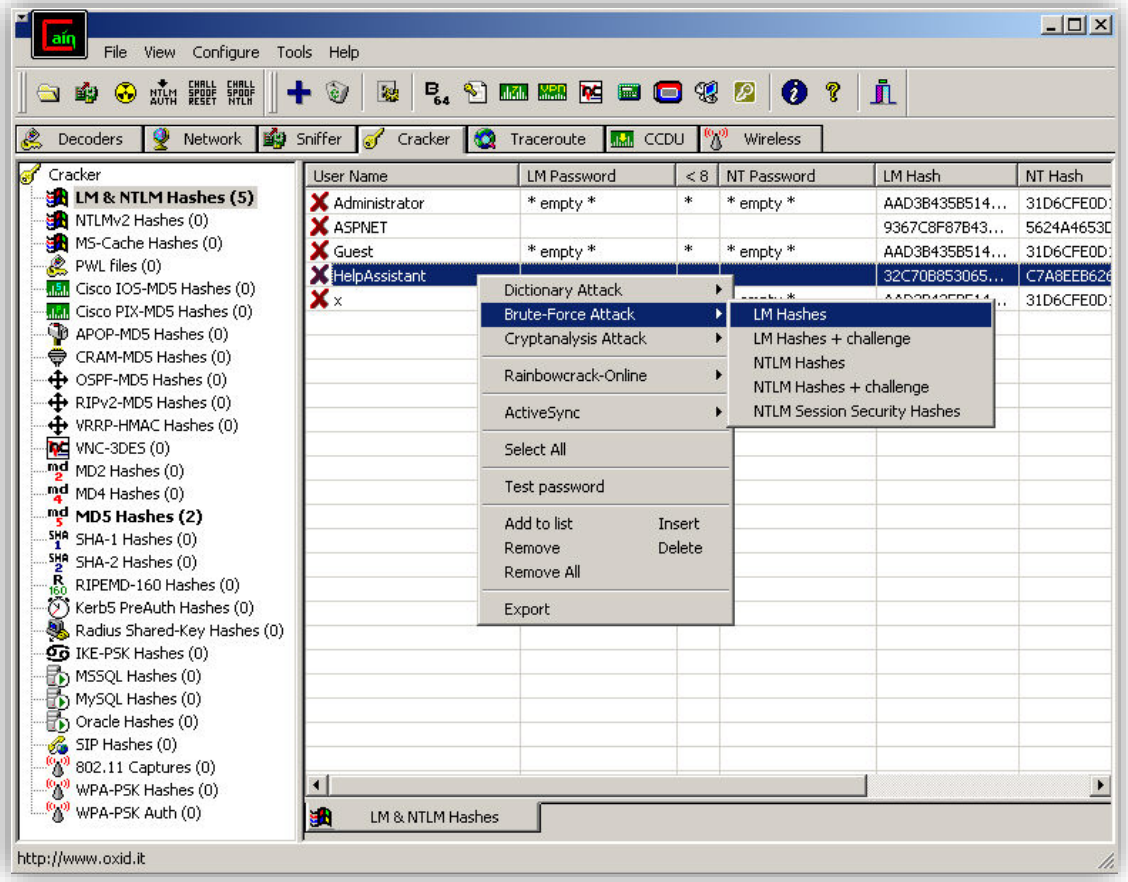
- Fast password cracker, multi-OS support
- Main purpose: detect weak Unix crypt(3) passwords
  - Has support for other hashes and cyphers (e.g. Windows LM)
- Highly optimized modules for different hash types and CPUs
- Latest versions include GPU support



Cain & Abel

http://www.oxid.it/cain.html

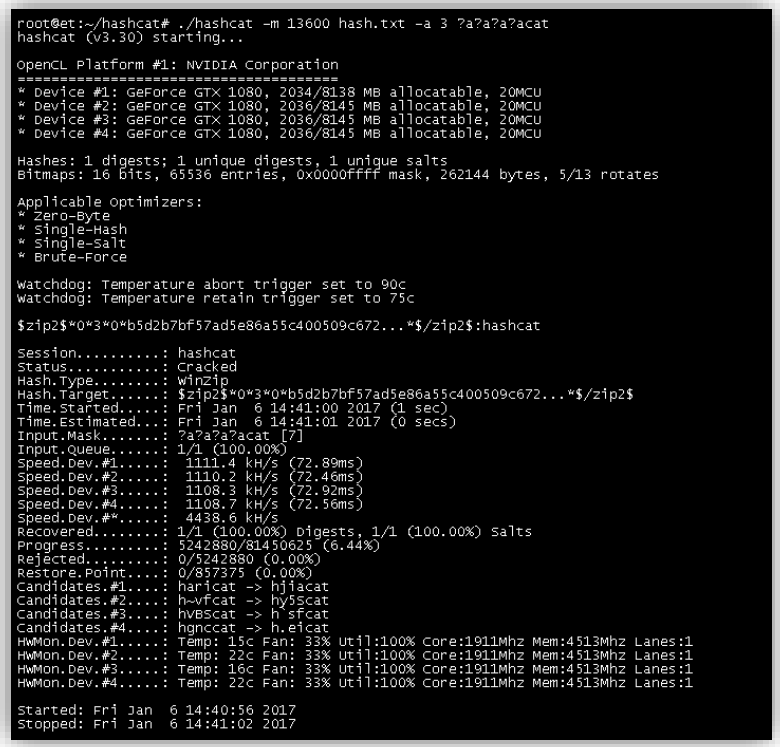
- "Password recovery tool" for Windows
  - Dictionary, brute-force, cryptanalysis
- Network sniffing
- ARP poisoning
- Recording VOIP conversations
- Recovering WiFi keys
- Uncovering cached passwords
- Not updated in a while ☹



Hashcat

https://hashcat.net

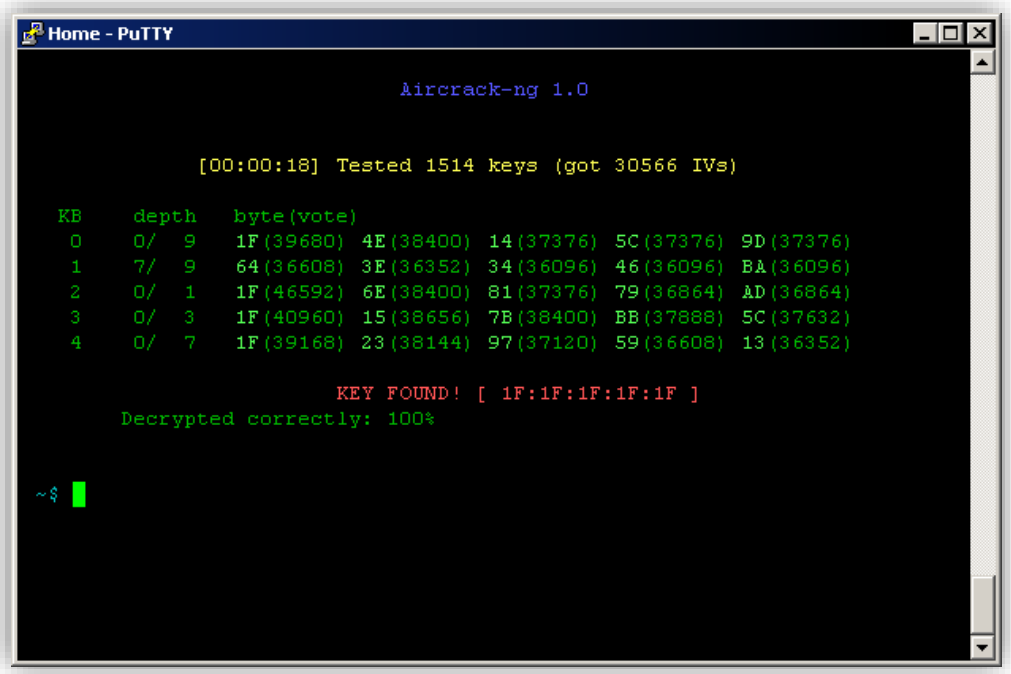
- One of the world's fastest password recovery tool
- GPU support (AMD, NVIDIA, Intel GPU), free, open source
- Integrated GPU thermal watchdog
- Multi-platform, 160+ hash types
- Built-in benchmarking engine
- Session & interactive pause/resume



Aircrack-ng

https://www.aircrack-ng.org/

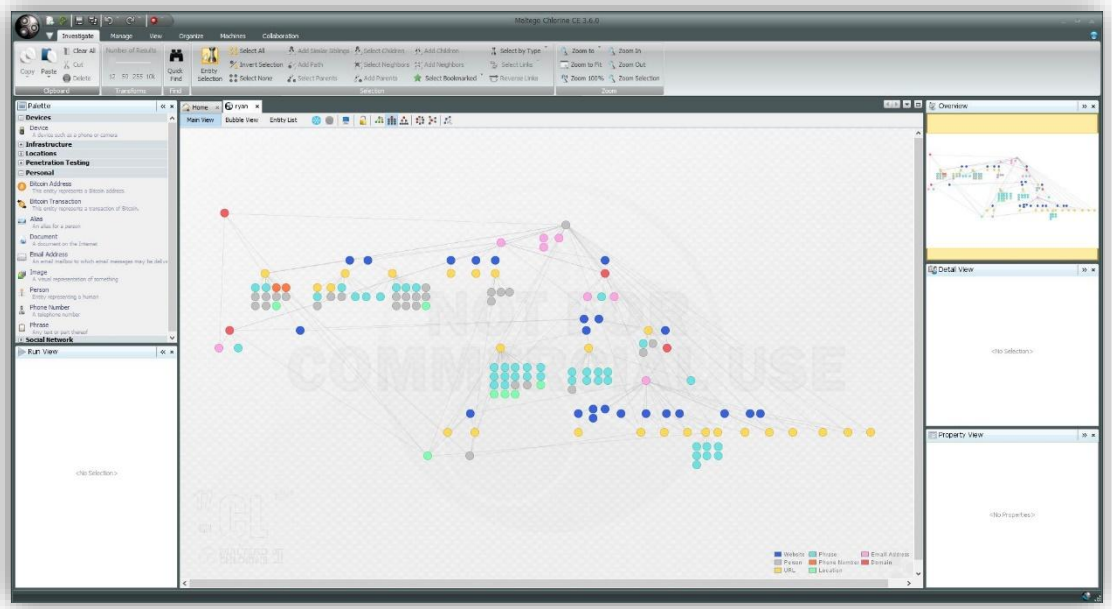
- Command-line WiFi network security assessment suite
- Monitoring & Packet Capture
- Attacking (replay, deauth, fake AP, packet injection)
- Cracking WEP and WPA PSK
- Testing WiFi cards



Maltego CE

https://www.maltego.com/pricing/

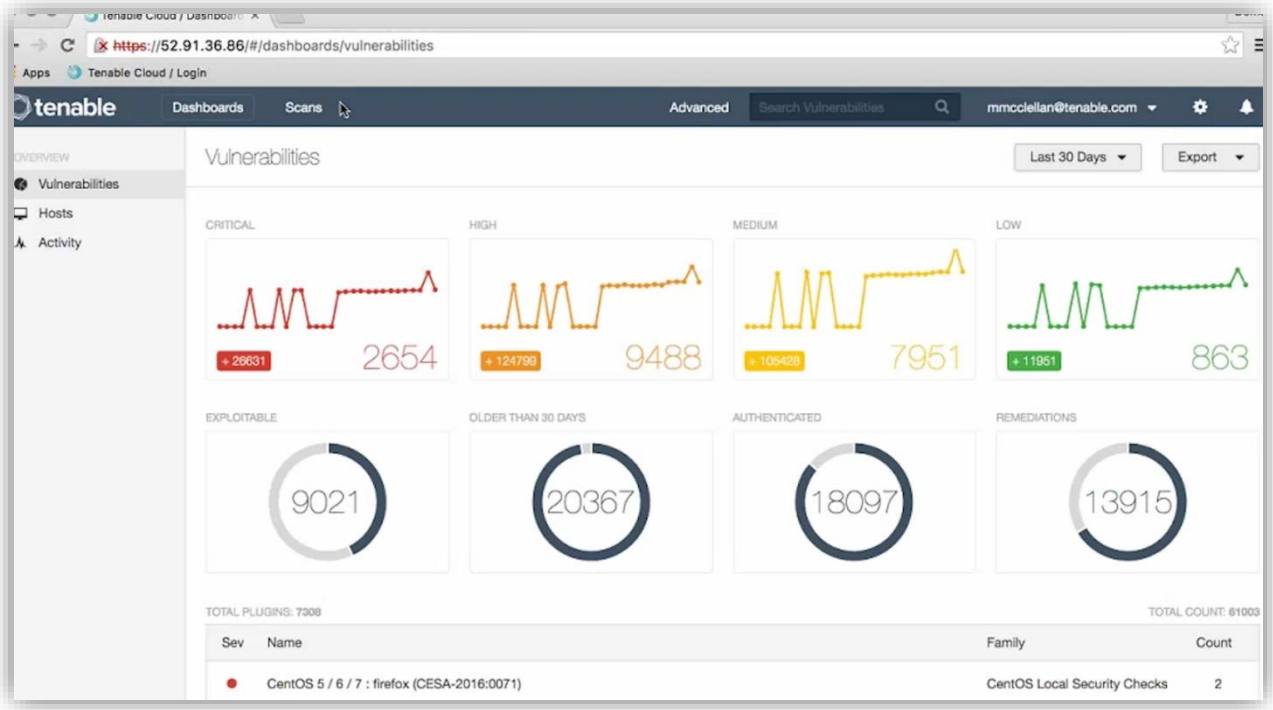
- Community Edition, ships with Kali
- Interactive data mining tool
- Renders directed graphs for link analysis
- Used in online investigation for finding relationships between pieces of info on the Internet
- Great for intelligence gathering
  - People, social networks, companies, organizations, websites, domains, DNS names, netblocks, IP addresses, affiliations, documents, files



Nessus

https://www.tenable.com/products/nessus/nessus-professional

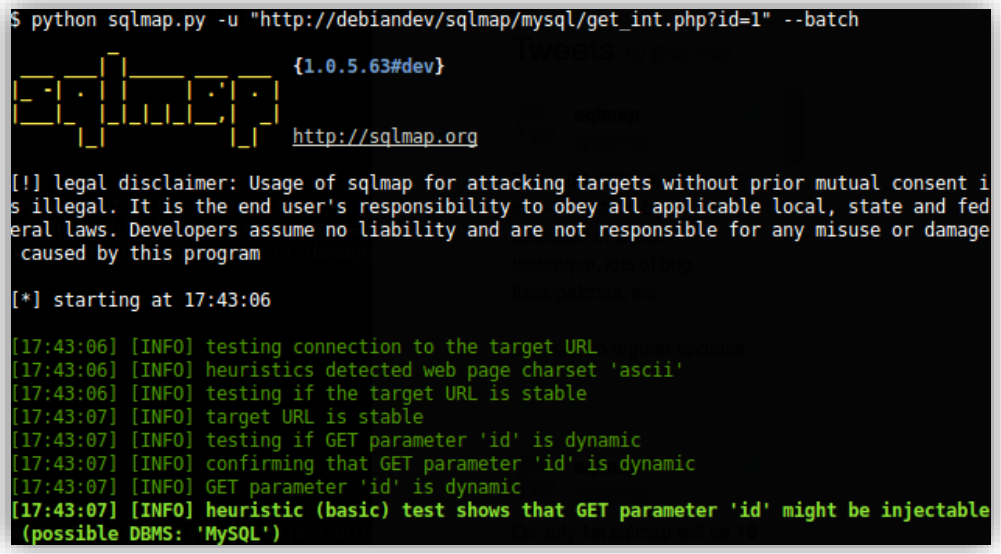
- Vulnerability scanning & management tool
- Ongoing scanning and reporting
  - Expandable with plugins (70.000+ right now)
- Free “Nessus Home” version
- One of the best price points on the market



Sqlmap

http://sqlmap.org/

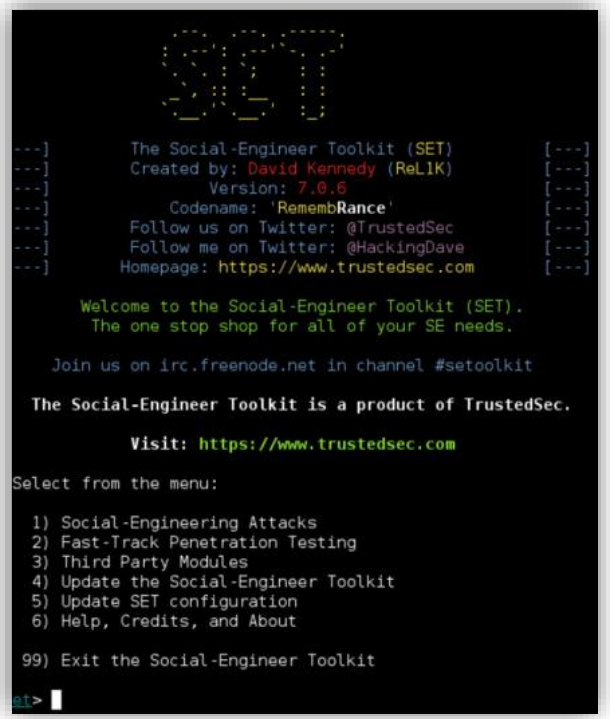
- Automatic SQL injection and DB takeover tool
- Detects and exploits SQL injection flaws
- Full support for MySQL, Oracle, PostgreSQL, Microsoft SQL Server, Microsoft Access, IBM DB2, SQLite, Firebird, Sybase, SAP MaxDB, HSQLDB and Informix
- Multiple injection techniques
  - Boolean-based blind, time-based blind, error-based, UNION query-based, stacked queries and out-of-band
- Enumerates users, password hashes, databases, roles, tables, columns



Social Engineering Toolkit

https://www.trustedsec.com/social-engineer-toolkit/

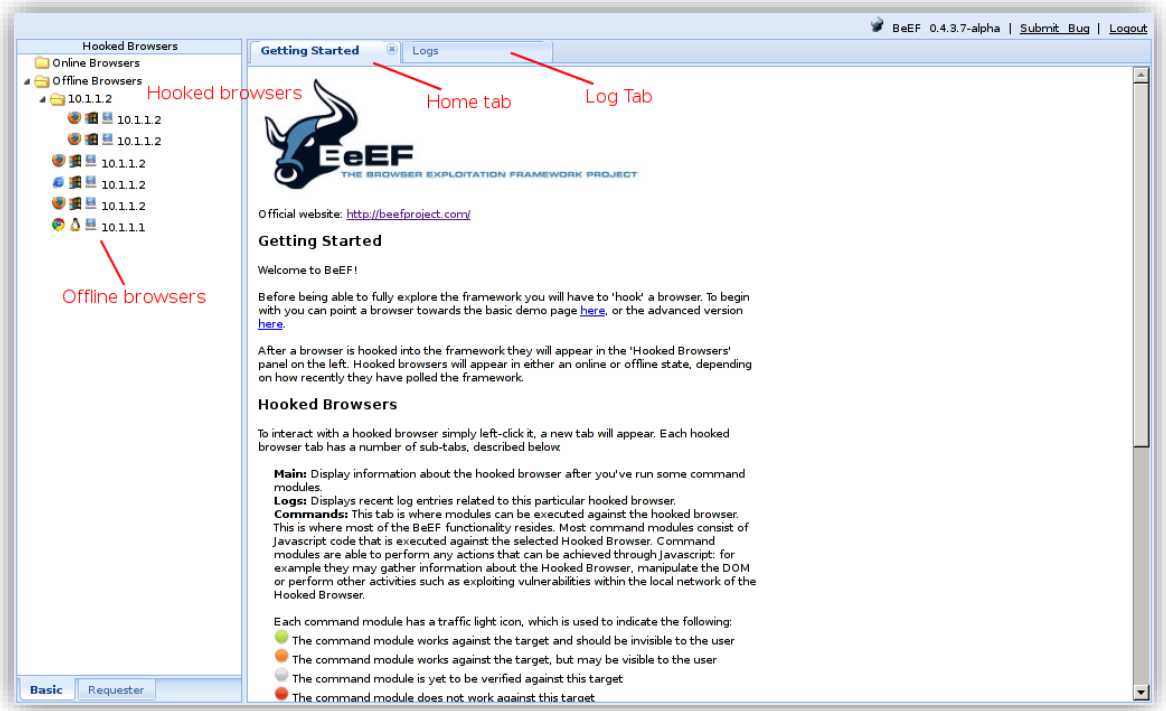
- Open-source Python-driven tool
- Used for penetration testing around Social Engineering
- Generates exploit-hiding web pages or emails
- Can use Metasploit payloads
- Included in Kali Linux



BeEF

http://beefproject.com/

- Browser Exploitation Framework Project
- Penetration testing tool focused on the browser
  - Uses client-side attack vectors
- Collecting of zombie browsers

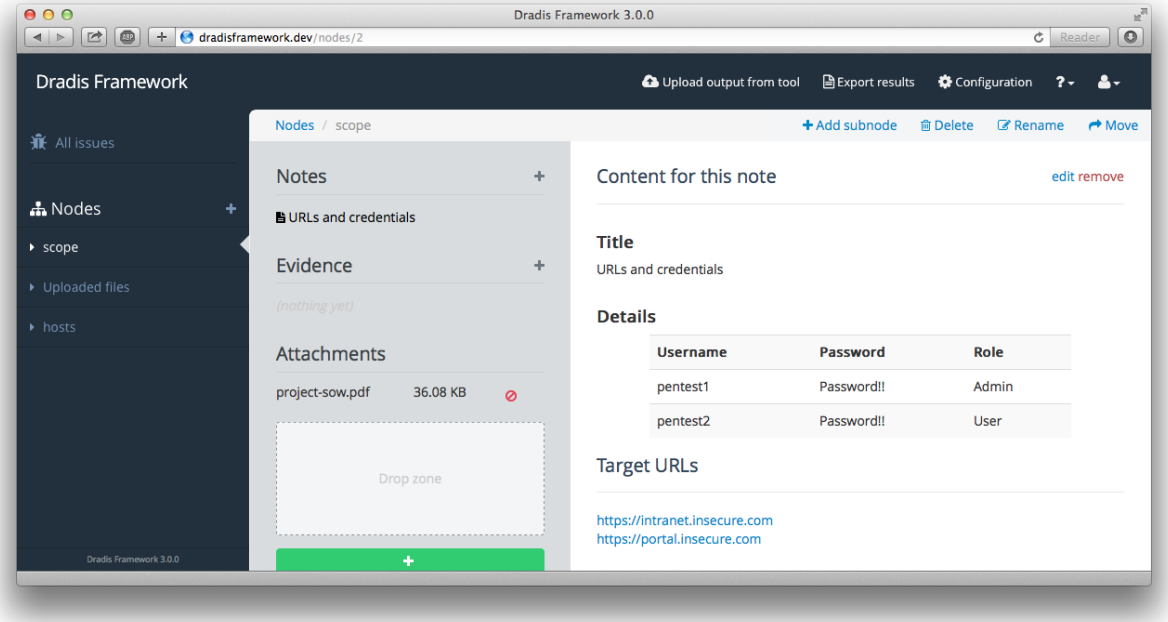


Dradis

https://dradis.com/ce/

Open source framework allowing information sharing during a penetration test via a self-contained web app

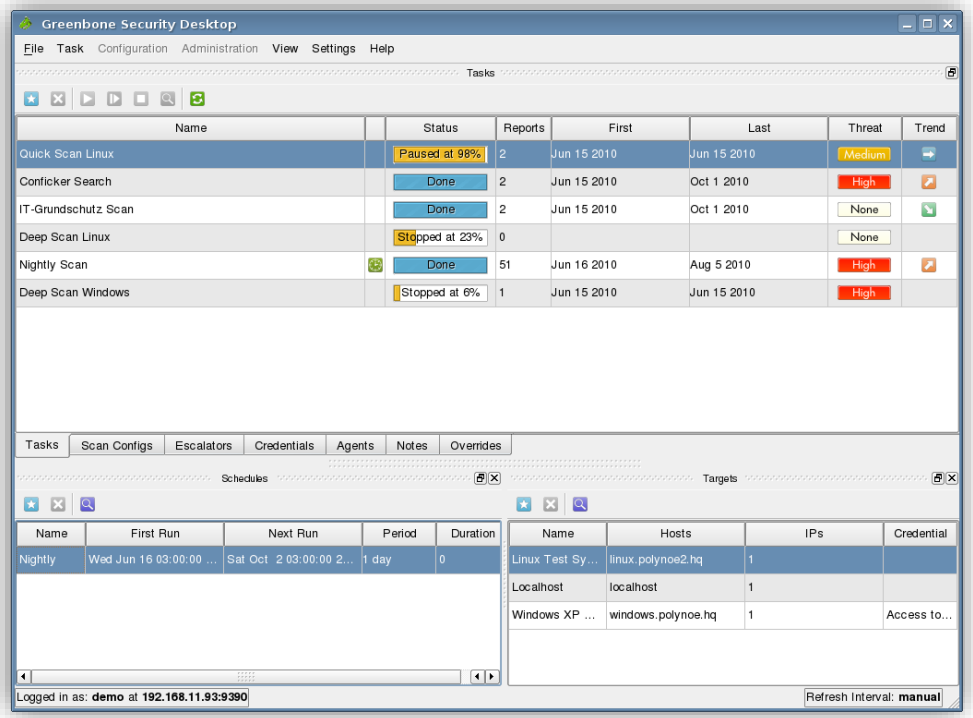
- Provides a centralized repository to keep track of work done
- Has plugins to read and collect output from a variety of tools (nmap, Burp Suite, nikto, etc.)



OpenVAS

http://www.openvas.org/

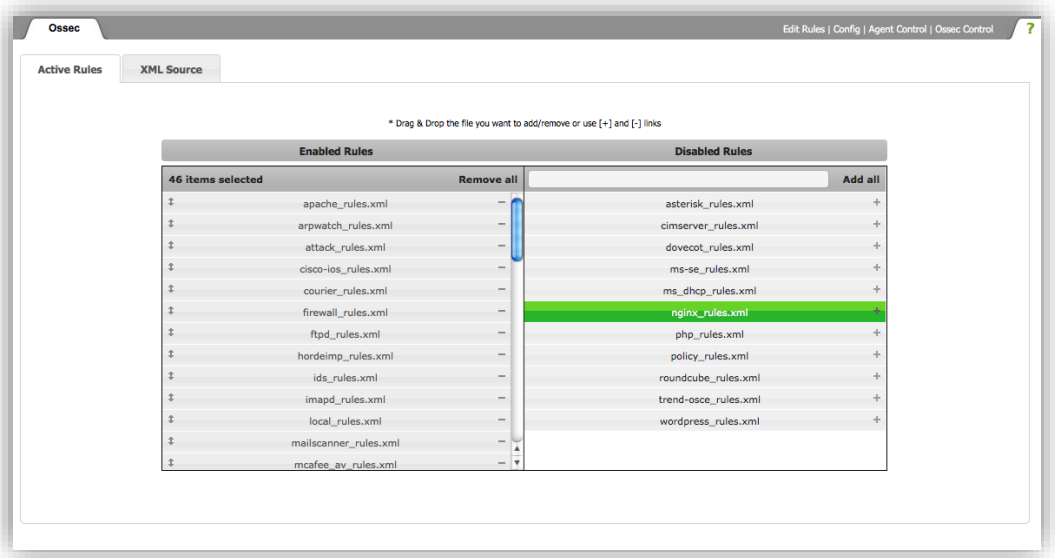
- Framework of several services and tools
  - Forked from the last free version of Nessus (2005)
- Open source vulnerability scanner and manager
  - Now Greenbone Community Edition



OSSEC

http://ossec.github.io/

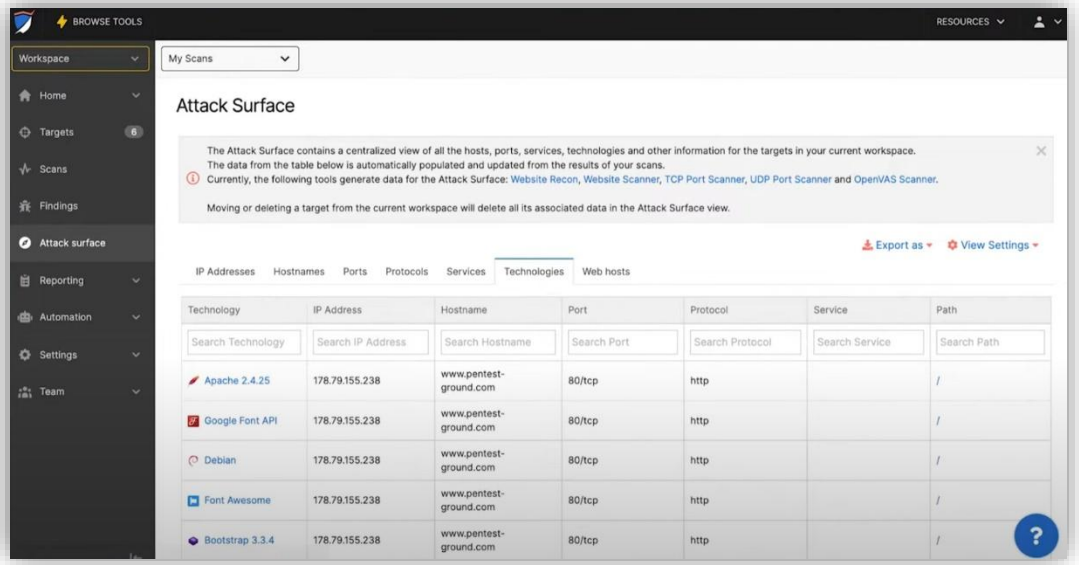
- Open Source HIDS Security
  - Host Intrusion Detection System for Linux, Solaris, AIX, BSD, Windows, Mac & VMWare ESX
- Helps customers meet compliance with PCI DSS
- Does log analysis, integrity checking, rootkit detection, time based alerting and active response



Pentest-Tools.com

https://pentest-tools.com/

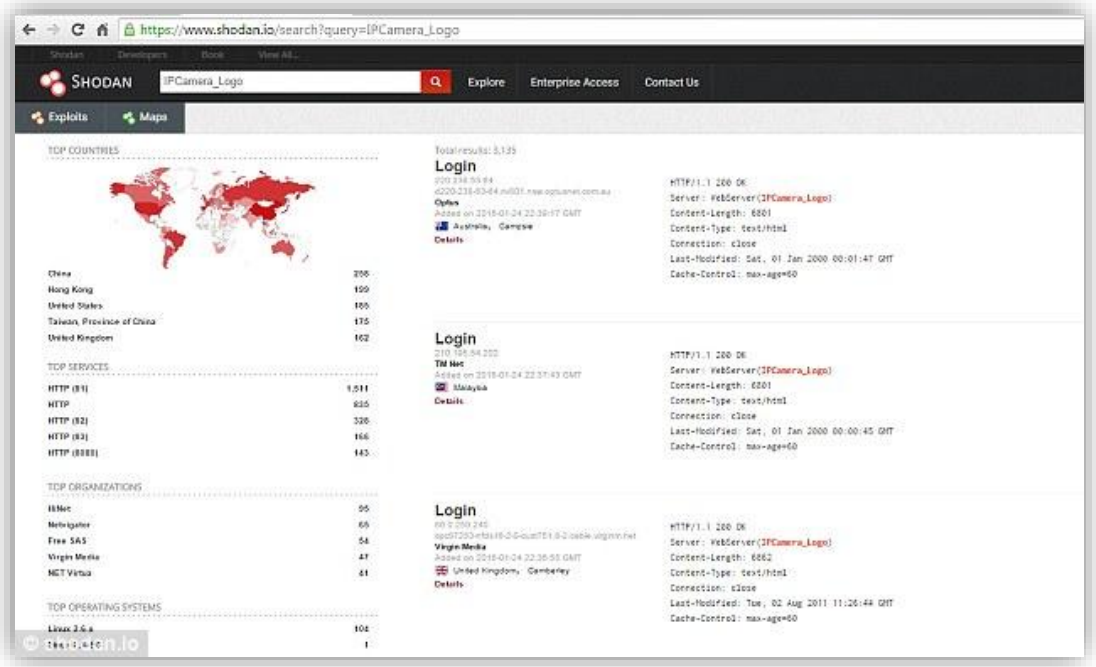
- A cloud-based solution for penetration testing and vulnerability assessments supporting the entire workflow of a security assessment
- It incorporates 20+ penetration testing tools and features dedicated to streamlining the process



Shodan.io

https://www.shodan.io/

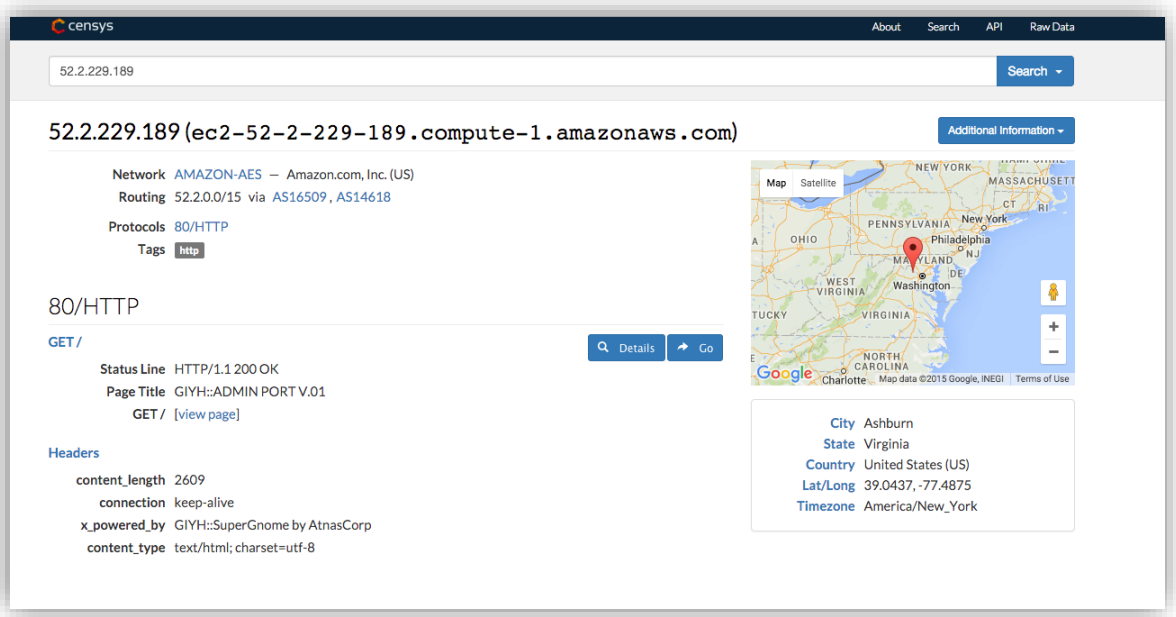
- The search engine for Security & Internet-connected devices
- Does regular scans of the entire Internet
- Maps devices and ports
- Performs banner grabbing
- Comes with browser plugins



Censys.io

https://www.censys.io/

- Search engine for internet devices
- Daily ZMap scans of the Internet
  - Hosts in the public IPv4 address space
  - Websites in the Alexa Top Million Domains
  - X.509 Certificates
- Lots of criteria for searches
  - https://www.censys.io/overview



The importance of Self Assessments

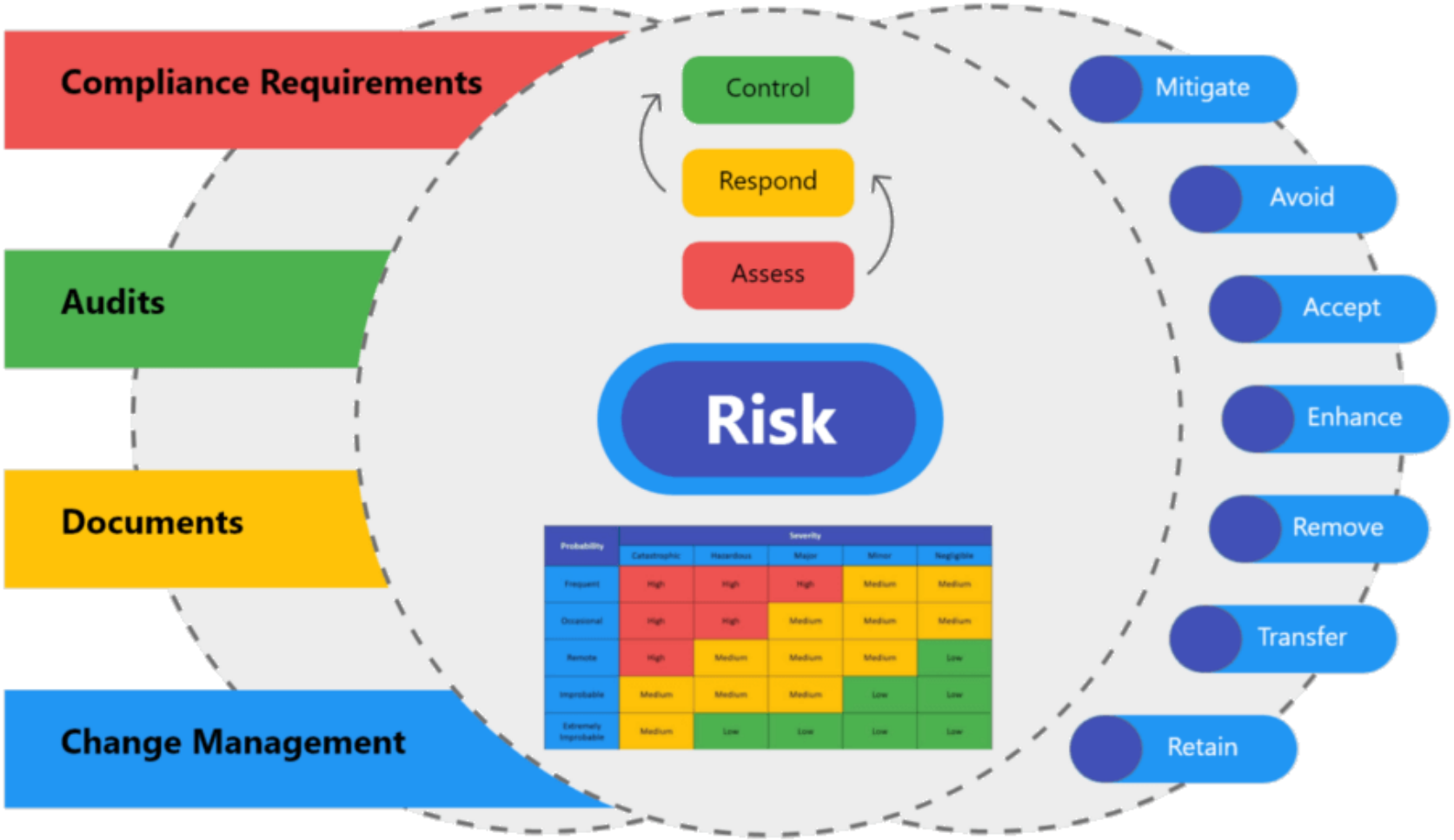
The need for a Cybersecurity Strategy

- What's your **Cybersecurity Maturity**?
  - Do you… have an antivirus/antimalware solution, continuously monitor the security posture, proactively identify and mitigate risks, regularly review and update security policies, etc.?
- Essential Components of a **Cybersecurity Strategy**
  - Governance, Risk Management & Compliance
  - Security Policies & Procedures
  - Identity & Access Control
  - Attack Surface Management
  - Data Protection
  - Incident Detection and Response
  - Vendor Security
  - Training and Awareness
  - Continuous Improvement
- Aligning the Cybersecurity Strategy with **Business Objectives**

Industry Standards and Best Practices

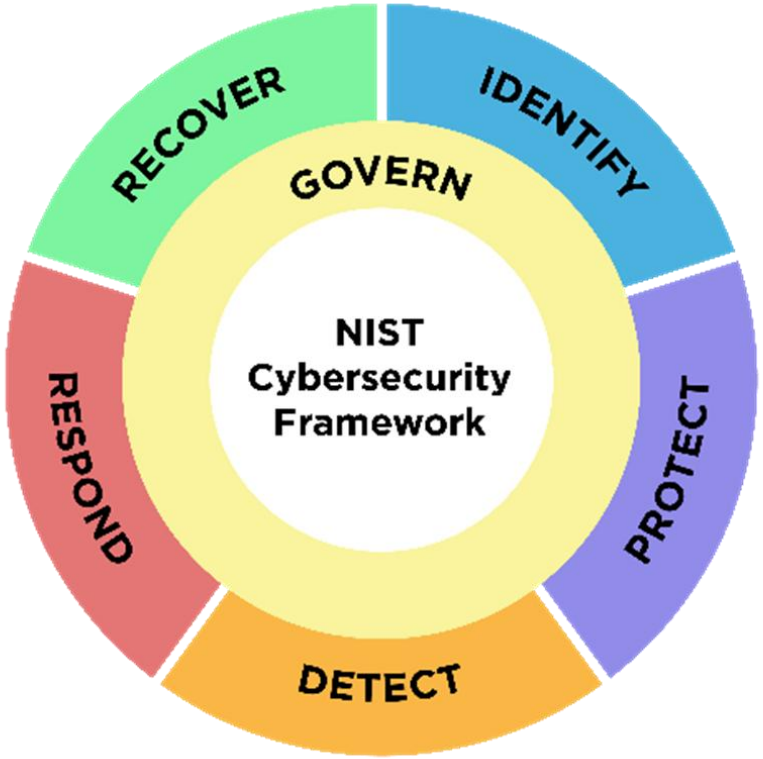
- **Why do Standards matter?** (e.g., NIST, ISO, COBIT)
  - Consistency & Efficiency
  - Proven Methods
  - Regulatory Compliance
  - Trust & Reputation
- **Importance of Best Practices** (Zero Trust, Attack Surface Management, etc.)
  - Never trust, always verify
- **Incorporating Standards** into the Cybersecurity Strategy
  - Understand the Standards
  - Gap Analysis
  - Action Plan
  - Implementation
  - Continuous Monitoring & Improvement

How is IT Risk usually handled?



NIST Cybersecurity Framework (CSF) 2.0

- Focuses on: Govern, Identify, Protect, Detect, Respond, Recover



Other Self Assessment Tools

- **ICT minimum standard**
  - The minimum standard for ICT (Information and Communication Technology) refers to the minimum level of proficiency or performance expected in the use of information and communication technologies
- **Software**: OWASP SAMM & ASVS
- **CSA STAR level 1**
  - Security, Trust, Assurance and Risk
  - https://cloudsecurityalliance.org/artifacts/star-level-1-security-questionnaire-caiq-v4/
- **CSA Cloud Controls Matrix**
  - CSA – Cloud Security Alliance
  - https://cloudsecurityalliance.org/research/cloud-controls-matrix/

Build an incident response plan

- No idea where to start?
  - Try NIST SP 800-61 (Computer Security Incident Handling Guide)

We’re nearly done ☺

Pentesting Methodologies and Tools

- Cybersecurity Fundamentals
- Anatomy of an Attack
- Security Assessments, Audits & Pentests
- Pentesting Methodologies
- Pentesting Tools
- Self Assessments

LINKEDIN

D3 CYBER

Thank you!

Contact: tudor.damian@d3cyber.eu | tudy.ro