

---

**USC CSci530**  
**Computer Security Systems**  
**Lecture notes**  
**Fall 2020 – Part II**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

---

# **CSci530: Security Systems**

## **Lecture 7&8 - October 7 & 21, 2022**

### **Untrusted Computing and Malicious Code**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

# Terminology

---

**Vulnerability** – A weakness in a system, program, procedure, or configuration that could allow an adversary to violate the intended policies of a system.

**Threat** – Tools or knowledge (capabilities) that are capable of exploiting a vulnerability to violate the intended policies of a system.

**Attack** – An attempt to exploit a vulnerability to violate the intended policies of a system.

**Compromise or intrusion** – The successful actions that violate the intended policies of a system.

# Terminology

---

**Trusted** – Parts of a system that we depend upon for the proper enforcement of policies, whether or not the code is free of vulnerabilities (almost all systems have vulnerabilities). - as compared with

**Trustworthy** – our belief that a system is free of vulnerabilities that could result in the violation the relevant security policies.

**Accreditation** – A statement by a third party that a system or software has been found to be trustworthy with respect to a particular set of policies and for a particular operational environment.

# **Incidents and Breaches**

---

**Penetration – A successful attack (intrusion) that exploits a vulnerability in the code base of a system or its configuration. The result will often be to install a subversion.**

**Denial of Service – An attack that prevents authorized access to a resource, by destroying a target or overwhelming it with undesired requests.**

**Subversion - An intentional change to the code base or configuration of a system that alters the proper enforcement of policy. This includes the installation of backdoors and other control channels in violation of the policy relevant to the system.**

**Subversion vectors – the methods by which subversions are introduced into a system. Often the vectors take the form of malicious code.**

# More Terminology

---

**Secure** – A system is secure if it correctly enforces a correctly stated policy for a system. A system can only be secure with respect to a particular set of policies and under a set of stated assumptions. There is no system that is absolutely secure.

**Trusted Computing Base** – That part of a system which if compromised affects the security of the entire system. One often unstated assumption made with respect to a secure system is that the TCB is correctly implemented and has not been compromised.

**Attack Surface** – The accumulation of all parts of a system that are exposed to an adversary against which the adversary can try to find and exploit a vulnerability that will render the system insecure (i.e. violate the security policies of the system).

# Attack Vectors

---

- **Trojan Horse**
  - Extra code added manually to web page, program, plugin, etc.
- **Viruses**
  - Self-propagating (on execution)
  - Contains a malicious payload.
- **Worms**
  - Self-propagating through process exploit.
  - Contains a malicious payload.
- **Penetration Tools (remote or local)**
  - Exploits vulnerabilities to violate policy
  - Injection, Overrun, Logic, other
- **Impersonation / Insider**

# **General Actions - Payloads**

---

- **Modification of data**
- **Spying - exfiltration**
- **Stepping off point for further attacks**
- **Advertising – and tracking interests**
- **Self Preservation - Rootkits**
- **Subversion**



# Malicious Actions

---

**Taken when attack vector is activated**

- **Malware propagation (Viruses and Worms)**
- **Subversion – Back doors, changes to software base**
  - **Spyware – Exfiltration of history, data, etc.**
  - **Zombies or bots or botnets – Remote control of system**
  - **Extortion - Ransomware – Destroy system or encrypt data and ask for ransom.**
  - **Bitcoin Miners**

---

# **CSci530: Security Systems**

## **Lecture 8 - October 21, 2022**

### **Untrusted Computing and Malicious Code (continued)**

**Dr. Clifford Neuman**

**University of Southern California**

**Information Sciences Institute**

# Defenses to Malicious Code

---

- **Detection**
  - Virus scanning
  - Intrusion Detection
- **Least Privilege**
  - Don't run as root
  - Separate users ID's
- **Isolation**
  - Mandatory controls on information flow
- **Sandboxing**
  - Limit what the program can do
- **Backup**
  - Keep something stable to recover

# **Categorizing Malicious Code**

---

## **How propagated**

- **Trojan Horses**
  - Embedded in useful program that others will want to run.
  - Covert secondary effect.
- **Viruses (an specialization of a Trojan horse)**
  - When program started will try to propagate itself.
- **Worms**
  - Exploits bugs to infect running programs.
  - Infection is immediate.

# Trojan Horses

---

- A desirable documented effect
  - Is why people run a program
- A malicious payload
  - An “undocumented” activity that might be counter to the interests of the user.
- Examples: Some viruses, much spyware.
- Issues: how to get user to run program.



# Trojan Horses

---

- **Software that doesn't come from a reputable source may embed trojans.**
- **Program with same name as one commonly used inserted in search path.**
- **Depending on settings, visiting a web site or reading email may cause program to execute.**

# Viruses

---

- **Resides within another program**
  - **Propagates itself to infect new programs (or new instances)**
- **May be an instance of Trojan Horse**
  - **Email requiring manual execution**
  - **Infected program becomes trojan**

# Viruses

---

- **Early viruses used boot sector**
  - **Instruction for booting system**
  - **Modified to start virus then system.**
  - **Virus writes itself to boot sector of all media.**
  - **Propagates by shared disks.**



# Viruses

---

- **Some viruses infect program**
  - **Same concept, on start program jumps to code for the virus.**
  - **Virus may propagate to other programs then jump back to host.**
  - **Virus may deliver payload.**

# **Viruses can be Spread by Email**

---

- **Self propagating programs**
  - **Use mailbox and address book for likely targets.**
  - **Mail program to targeted addresses.**
  - **Forge sender to trick recipient to open program.**
  - **Exploit bugs to cause auto execution on remote site.**
  - **Trick users into opening attachments.**

# Viruses Phases

---

- **Insertion Phase**
  - How the virus propagates
- **Execution phase**
  - Virus performs other malicious action
- **Virus returns to host program**

# **Analogy to Real Viruses**

---

- **Self propagating**
- **Requires a host program to replicate.**
- **Similar strategies**
  - **If deadly to start won't spread very far – it kills the host.**
  - **If infects and propagates before causing damage, can go unnoticed until it is too late to react.**

# How Viruses Hide

---

- Encrypted in random key to hide signature.
- Polymorphic viruses changes the code on each infection.
- Some viruses cloak themselves by trapping system calls.

# Macro Viruses

---

- Code is interpreted by common application such as word, excel, postscript interpreter, etc.
- May be virulent across architectures.

# **Worms**

---

- **Propagate across systems by exploiting vulnerabilities in programs already running.**
  - **Buffer overruns on network ports**
  - **Does not require user to “run” the worm, instead it seeks out vulnerable machines.**
  - **Often propagates server to server.**
  - **Can have very fast spread times.**

# Delayed Effect

---

- Malicious code may go undetected if effect is delayed until some external event.
  - A particular time
  - Some occurrence
  - An unlikely event used to trigger the logic.



# **Zombies/Bots**

---

- **Machines controlled remotely**
  - **Infected by virus, worm, or trojan**
  - **Can be contacted by master**
  - **May make calls out so control is possible even through firewall.**
  - **Often uses IRC for control.**

# **Spyware**

---

- **Infected machine collect data**
  - **Keystroke monitoring**
  - **Screen scraping**
  - **History of URL's visited**
  - **Scans disk for credit cards and password.**
  - **Allows remote access to data.**
  - **Sends data to third party.**

## **Some Spyware Local**

---

- **Might not ship data, but just uses it**
  - **To pop up targeted ads**
  - **Spyware writer gets revenue for referring victim to merchant.**
  - **Might rewrite URL's to steal commissions.**
- **Superfish**

# **Theory of Malicious Code**

---

- **Can not detect a virus by determining whether a program performs a particular activity.**
  - **Reduction from the Halting Problem**
- **But can apply heuristics**

# Defenses to Malicious Code

---

- **Detection**
  - **Signature based**
  - **Activity based**
- **Prevention**
  - **Prevent most instances of memory used as both data and code**

# Defenses to Malicious Code

---

- **Sandbox**
  - Limits access of running program
  - So doesn't have full access or even users access.
- **Detection of modification**
  - Signed executables
  - Tripwire or similar
- **Statistical detection**

# **Root Kits - Subversion**

---

- **Hide traces of infection or control**
  - **Intercept systems calls**
  - **Return false information that hides the malicious code.**
  - **Returns fall information to hide effect of malicious code.**
  - **Some root kits have countermeasures to attempts to detect the root kits.**
  - **Blue pill makes itself hyper-root**

# **Best Detection is from the Outside**

---

- **Platform that is not infected**
  - **Look at network packets using external device.**
  - **Mount disks on safe machine and run detection on the safe machine.**
  - **Trusted computing can help, but still requires outside perspective**



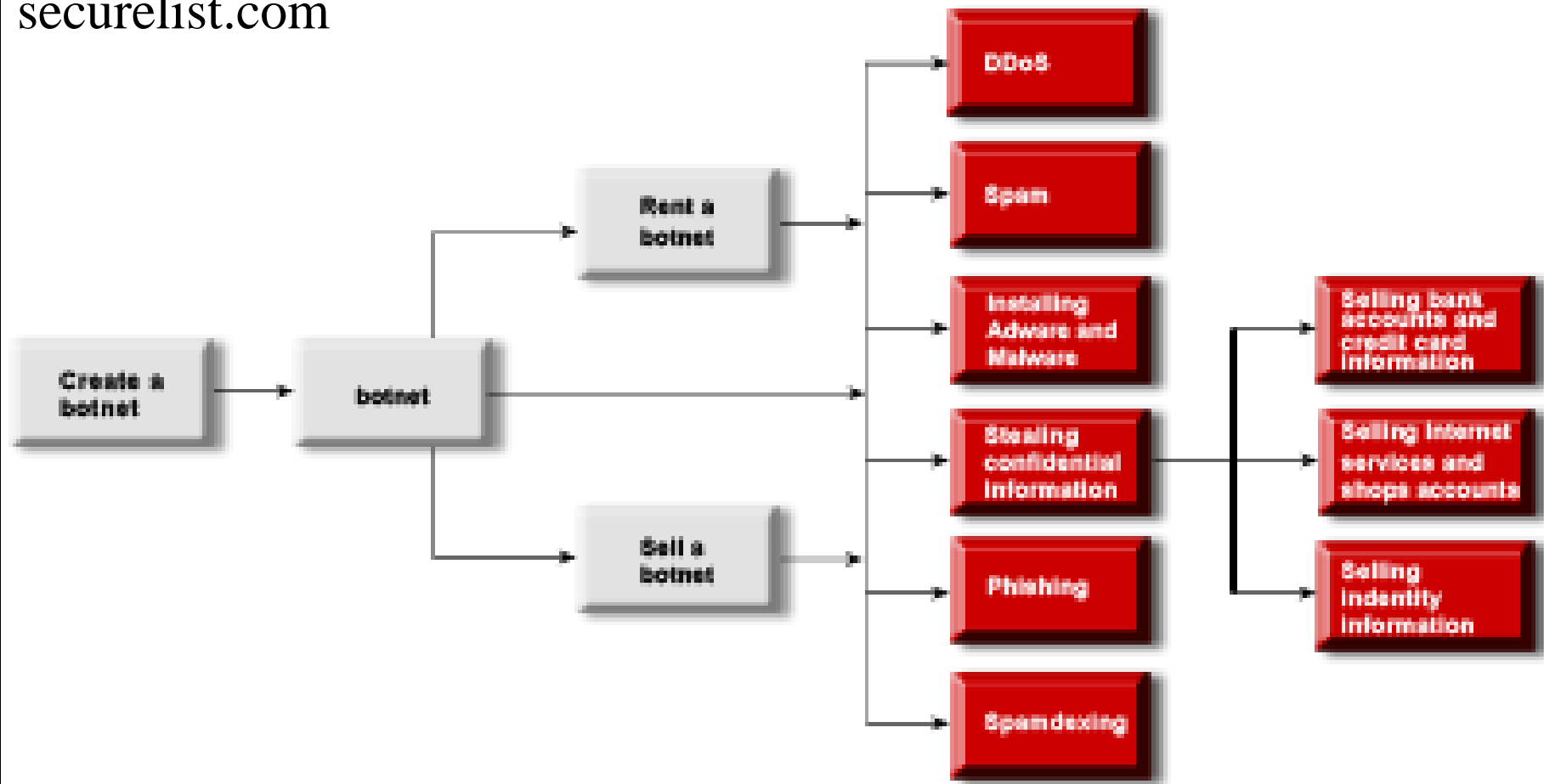
# **Economics of Malicious Code**

---

- **Controlled machines for sale**
- **“Protection” for sale**
- **Attack software for sale**
- **Stolen data for sale**
- **Intermediaries used to convert online balances to cash.**
  - **These are the pawns and the ones that are most easily caught**

# Economics of Malicious Code

Source: Yuri Namestnikov Money stream  
securelist.com



# **Economics of Adware and Spam**

---

- **Might not ship data, but just uses it**
  - **To pop up targeted ads**
  - **Spyware writer gets revenue for referring victim to merchant.**
  - **Might rewrite URL's to steal commissions.**

# New Monitization Technique

---

- Malware mining of bitcoins – Slashdot – 9/25/2017
  - Two Showtime domains are currently loading and running Coinhive, a JavaScript library that mines Monero using the CPU resources of users visiting Showtime's websites. The two domains are showtime.com and showtimeanytime.com, the latter being the official URL for the company's online video streaming service. It is unclear if someone hacked Showtime and included the mining script without the company's knowledge. Showtime did not respond to a request for comment, but it could be an experiment as the setThrottle value is 0.97, meaning the mining script will remain dormant for 97% of the time.

---

# **CSci530: Security Systems**

## **Lecture 9 – October 28, 2022**

### **Countermeasures**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

# Intrusion Prevention

---

- **A Marketing buzzword – for better than detection**
- **In Reality – General Good practices fall in this category**
  - **We will discuss network architectures**
  - **We will discuss Firewalls**
  - **This lecture is about this kind of intrusion prevention**
- **Intrusion detection (next week)**
  - **Term used for networks**
  - **But applies to host as well**
    - **Tripwire**
    - **Virus checkers**
- **Intrusion response (part now, part next week)**
  - **Evolving area**
    - **Anti-virus tools have a response component**
    - **Can be tied to policy tools**

# Architecture: A first step

---

- **Understand your applications**  
**Information Flow:**
  - What is to be protected
  - Against which threats
  - Who needs to access which apps
  - From where must they access it
- **Do all this before you invest in the latest products that salespeople will say will solve your problems.**

# What is to be protected

---

- **Is it the service or the data?**
  - **Data is protected by making it less available**
  - **Services are protected by making them more available (redundancy)**
  - **The hardest cases are when one needs both.**



# Classes of Data

---

- **Decide on multiple data classes**
  - **Public data**
  - **Customer data**
  - **Corporate data**
  - **Highly sensitive data**  
**(not total ordering)**
- **These will appear in different parts of the network**

# Classes of Users

---

- **Decide on classes of users**
  - **Based on the access needed to the different classes of data.**
- **You will architect your system and network to enforce policies at the boundaries of these classes.**
  - **You will place data to make the mapping as clean as possible.**
- **You will manage the flow of data**

# Example

---

- Where will you place your companies public web server, so that you can be sure an attacker doesn't hack your site and modify your front page?
- Where will you place your customer's account records so that they can view them through the web?
  - How will you get updates to these servers?

# Other Practices

---

- **Run Minimal Systems**
  - Don't run services you don't need
  - The Principle of least privilege
- **Patch Management**
  - Keep your systems up to date on the current patches
  - But don't blindly install all patches right away either (possible subversion vector)
- **Account management**
  - Strong passwords, delete accounts when employees leave, etc.
- **Don't rely on passwords alone**

---

# **CSci530: Security Systems**

## **Lecture 9 – October 28, 2022**

### **Countermeasures**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

# How to think of Firewalled Network

---

**Crunchy on the outside.**

**Soft and chewy on the inside.**

**– Bellovin and Merrit**

# Firewalls

---

- **Packet filters**
  - **Stateful packet filters**
    - **Common configuration**
- **Application level gateways or Proxies**
  - **Common for corporate intranets**
- **Host based software firewalls**
  - **Manage connection policy**
- **Virtual Private Networks**
  - **Tunnels between networks**
  - **Relationship to IPsec**

# Packet Filter

---

- **Most common form of firewall and what one normally thinks of**
- **Rules define what packets allowed through**
  - **Static rules allow packets on particular ports and to and from outside pairs of addresses.**
  - **Dynamic rules track destinations based on connections originating from inside.**
  - **Some just block inbound TCP SYN packets**



# Network Address Translation

---

- **Many home firewalls today are NAT boxes**
  - **Single address visible on the outside**
  - **Private address space (net 10, 192.168) on the inside.**
- **Hides network structure, hosts on inside are not addressable.**
  - **Box maps external connections established from inside back to the private address space.**
- **Servers require persistent mapping and manual configuration.**
  - **Many protocols, including attacks, are designed to work through NAT boxes.**

# Application FW or Proxies

---

- **No direct flow of packets**
  - Instead, connect to proxy with application protocol.
  - Proxy makes similar request to the server on the outside.
- **Advantage**
  - Can't hide attacks by disguising as different protocol.
  - But can still encapsulate attack.
- **Disadvantage**
  - Can't do end to end encryption or security since packets must be interpreted by the proxy and recreated.

# Host Based Firewalls

---

- **Each host has its own firewall.**
  - **Closer to the data to be protected**
  - **Avoids the chewy on the inside problem in that you still have a boundary between each machine and even the local network.**
- **Problems**
  - **Harder to manage**
  - **Can be subverted by malicious applications.**

# Embedded and Distributed FW

---

- **Embedded Firewalls**
  - Implemented on hardware cards (firmware)
  - Better protected against subversion
  - Requires protected management component.
- **Distributed Firewalls**
  - Policy managed from central location
  - Flows managed by individual host, embedded, or appliance based firewalls.
  - Coordinated view of system policies.

# Virtual Private Networks

---

- **Extend perimeter of firewalled networks**
  - **Two networks connected**
  - **Encrypted channel between them**
  - **Packets in one zone tunneled to other and treated as originating within same perimeter.**
- **Extended network can be a single machine**
  - **VPN client tunnels packets**
  - **Gets address from VPN range**
  - **Packets encrypted in transit over open network**

# IPSec

---

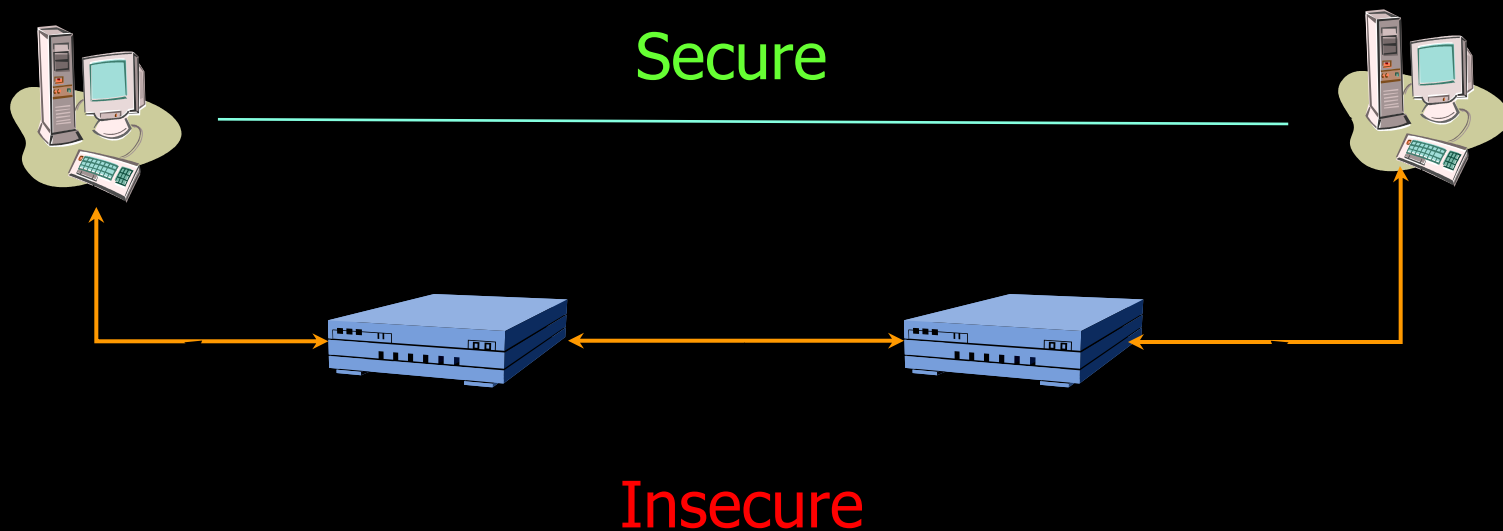
- **IP Security (IPsec) and the security features in IPv6 essentially move VPN support into the operating system and lower layers of the protocol stack.**
- **Security is host to host, or host to network, or network to network as with VPN's**
  - **Actually, VPN's are rarely used host to host, but if the network had a single host, then it is equivalent.**

## **IPSec Goals**

---

- **Authentication of hosts**
  - *Verify the source of IP packets*
  - *Prevention of replays*
- **Verify integrity of packets**
  - Through use of hashes and cryptography
- **Ensure confidentiality of packets**
  - **Protect the payload**

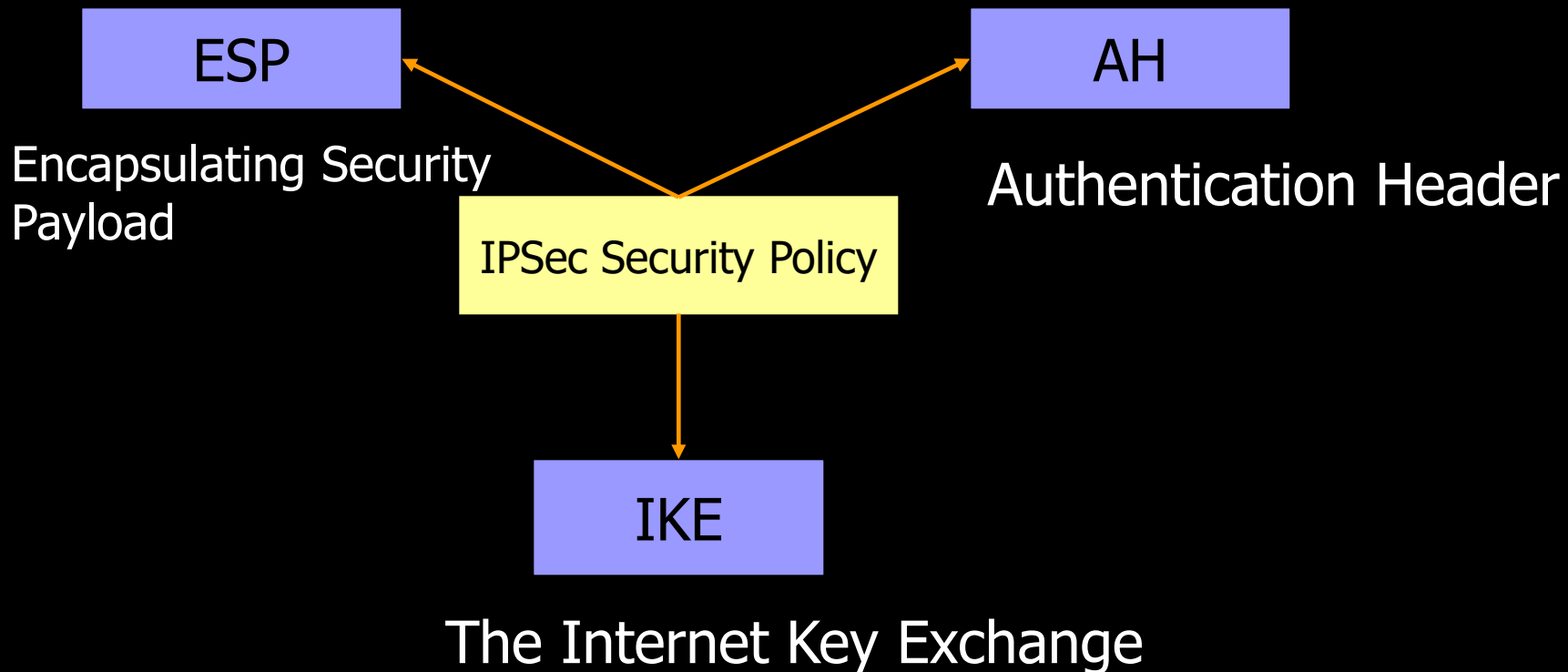
# The IPSec Security Model





# IPSec Architecture

---

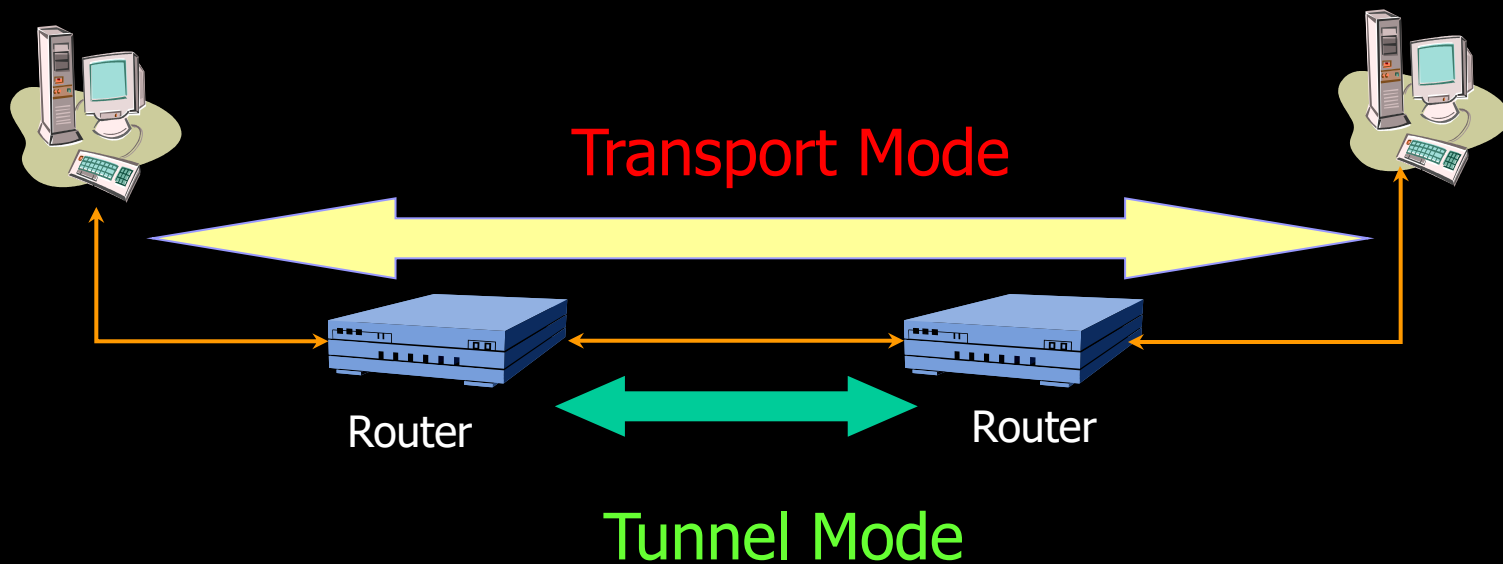


# IPSec Architecture

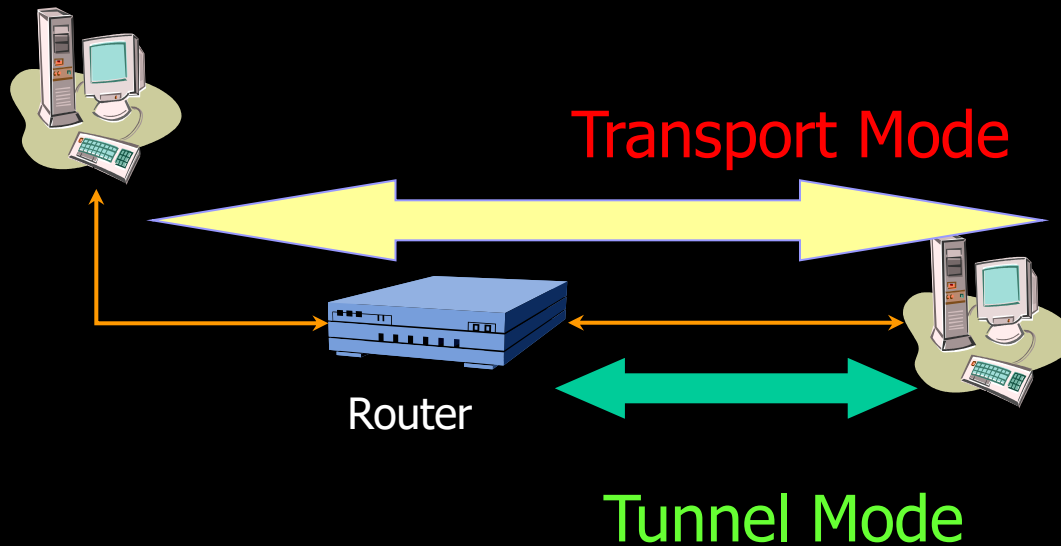
---

- **IPSec provides security in three situations:**
  - **Host-to-host, host-to-gateway and gateway-to-gateway**
- **IPSec operates in two modes:**
  - ***Transport mode* (for end-to-end)**
  - ***Tunnel mode* (for VPN)**

# IPsec Architecture



# IPsec Architecture



# Various Packets

---

Original

IP header	TCP header	data
-----------	------------	------

Transport  
mode

IP header	IPSec header	TCP header	data
-----------	--------------	------------	------

Tunnel  
mode

IP header	IPSec header	IP header	TCP header	data
-----------	--------------	-----------	------------	------

# Authentication Header (AH)

---

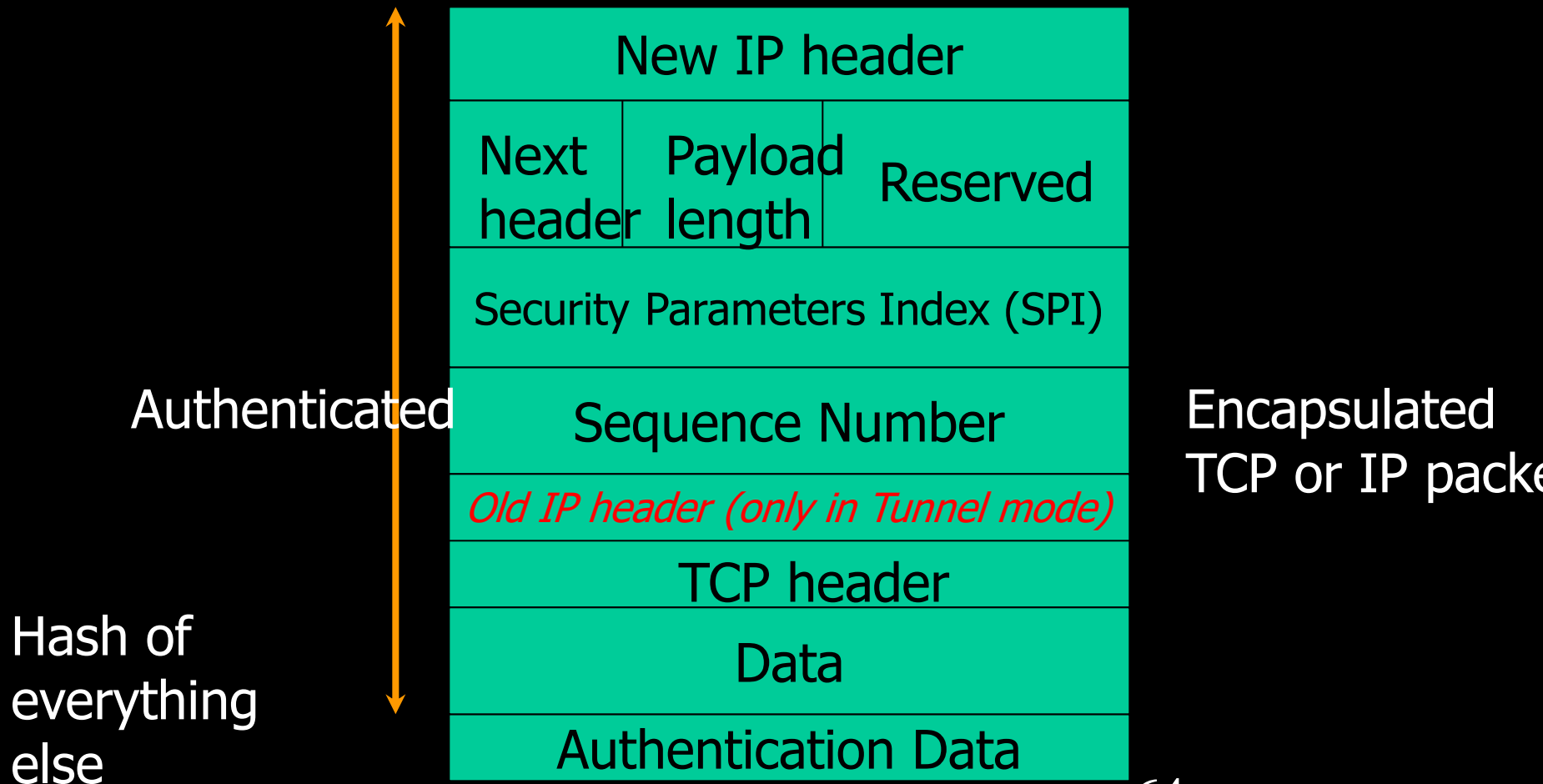
- Provides source authentication
  - Protects against source spoofing
- Provides data integrity
- Protects against replay attacks
  - Use monotonically increasing sequence numbers
  - Helps Protect against dos attacks
- **NO protection for confidentiality!**

## AH Details

---

- Use 32-bit monotonically increasing sequence number to avoid replay attacks
- Use cryptographically strong hash algorithms to protect data integrity (96-bit)
  - Use symmetric key cryptography
  - HMAC-SHA-96, HMAC-MD5-96

# AH Packet Details





# **Encapsulating Security Payload (ESP)**

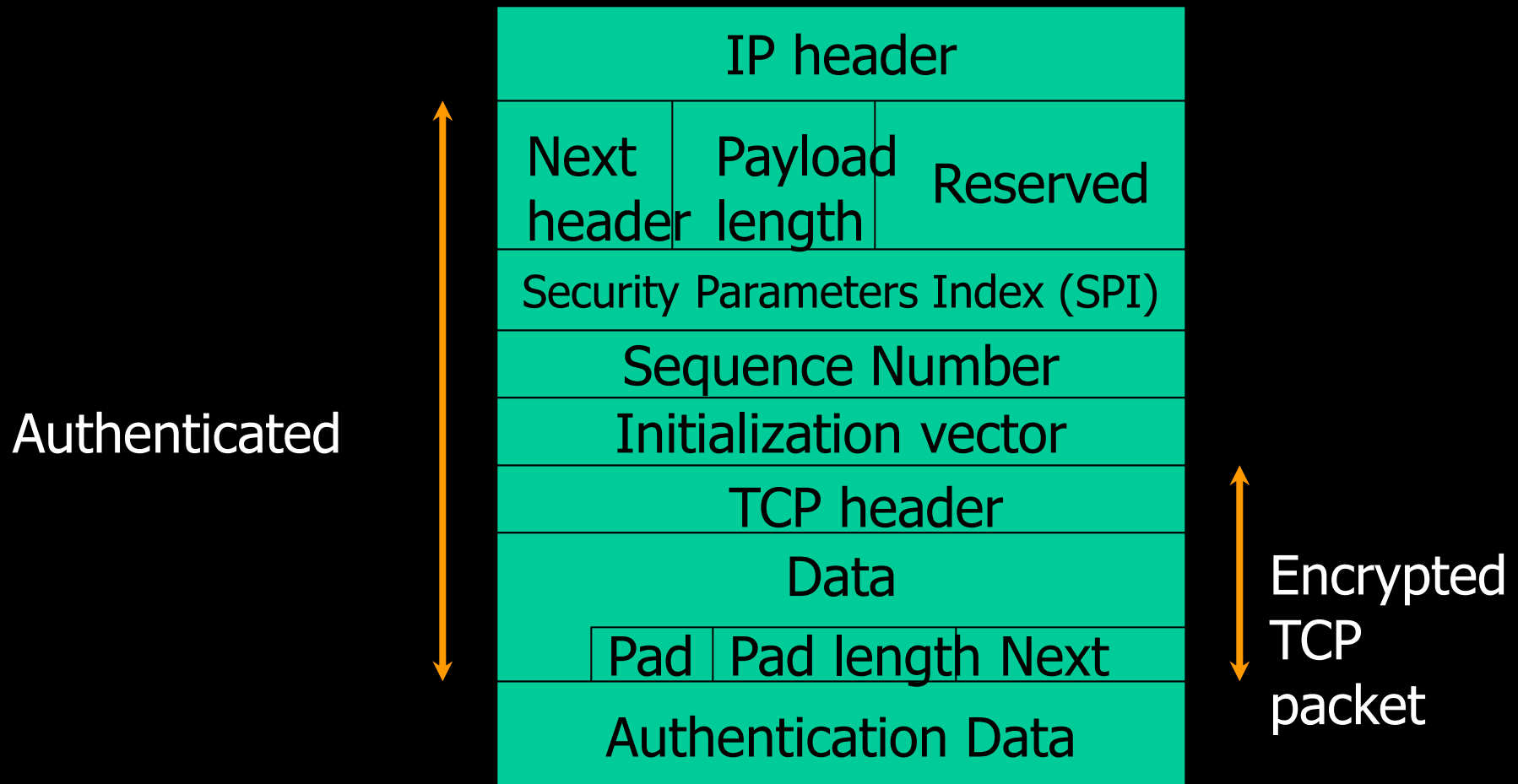
- Provides all that AH offers, and
- in addition provides **data confidentiality**
  - Uses symmetric key encryption

# ESP Details

---

- Same as AH:
  - Use 32-bit sequence number to counter replaying attacks
  - Use integrity check algorithms
- Only in ESP:
  - Data confidentiality:
    - Uses symmetric key encryption algorithms to encrypt packets

# ESP Packet Details



# Internet Key Exchange (IKE)

---

- Exchange and negotiate security policies
- Establish security sessions
  - Identified as *Security Associations*
- Key exchange
- Key management
- Can be used outside IPsec as well

# IPsec/IKE Acronyms

---

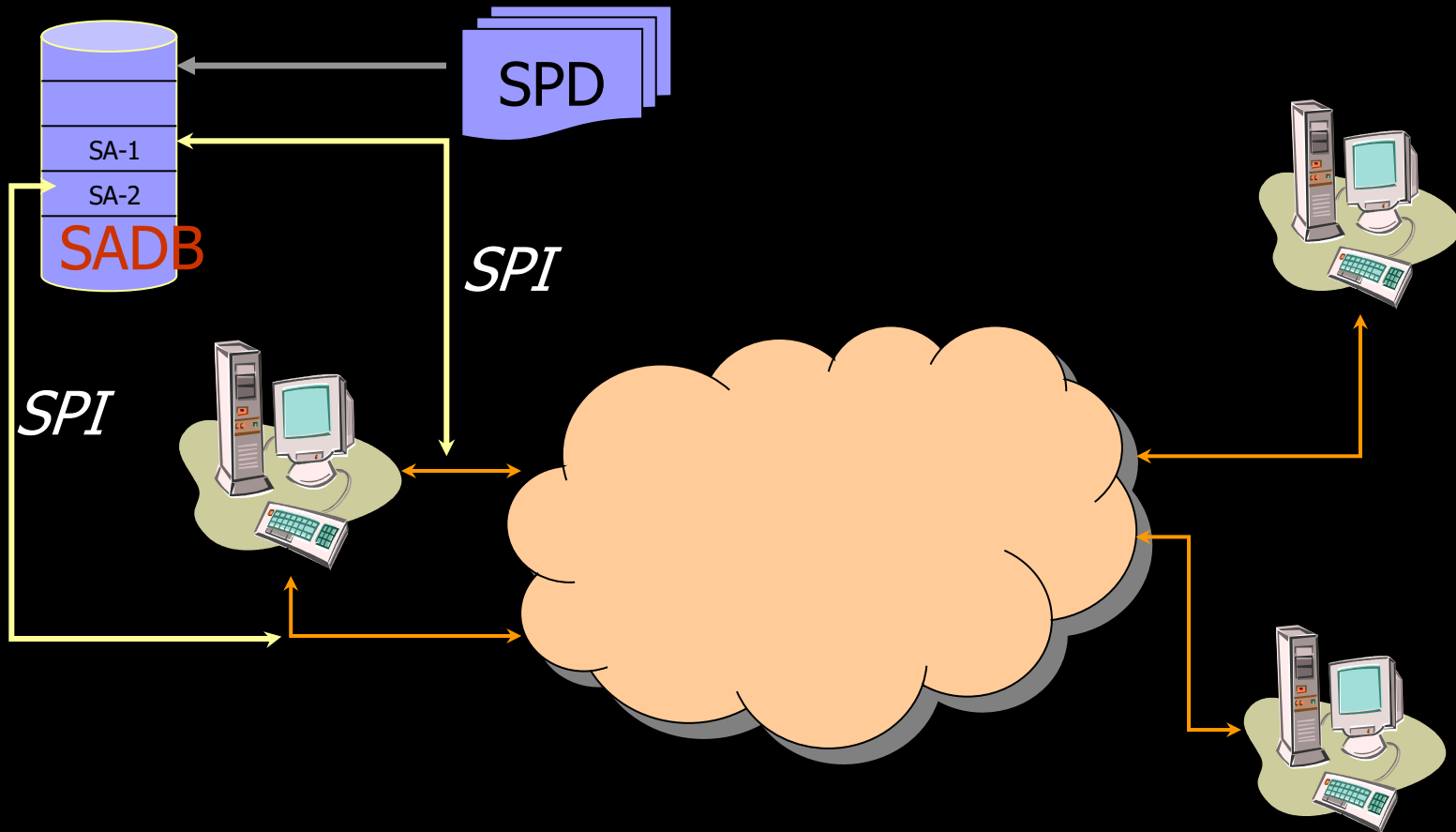
- **Security Association (SA)**
  - Collection of attribute associated with a connection
  - Is *asymmetric!*
    - One SA for inbound traffic, another SA for outbound traffic
    - Similar to ciphersuites in SSL
- **Security Association Database (SADB)**
  - A database of SAs

# **IPsec/IKE Acronyms**

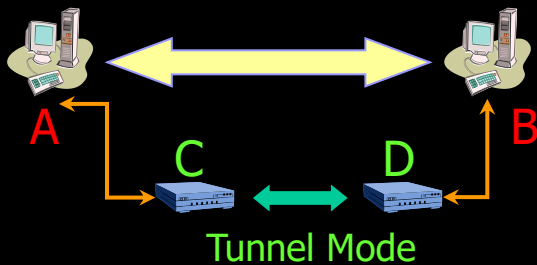
---

- **Security Parameter Index (SPI)**
  - A unique index for each entry in the SADB
  - Identifies the SA associated with a packet
- **Security Policy Database (SPD)**
  - Store policies used to establish SAs

# How They Fit Together



# SPD and SADB Example



A's SPD

From	To	Protocol	Port	Policy
A	B	Any	Any	AH[HMAC-MD5]

A's SADB

From	To	Protocol	SPI	SA Record
A	B	AH	12	HMAC-MD5 key

From	To	Protocol	Port	Policy	Tunnel Dest
A <sub>sub</sub>	B <sub>sub</sub>	Any	Any	ESP[3DES]	D

C's SPD

From	To	Protocol	SPI	SA Record
A <sub>sub</sub>	B <sub>sub</sub>	ESP	14	3DES key

C's SADB



# How It Works

---

- IKE operates in two phases
  - **Phase 1:** negotiate and establish an auxiliary end-to-end secure channel
    - Used by subsequent phase 2 negotiations
    - Only established once between two end points!
  - **Phase 2:** negotiate and establish custom secure channels
    - Occurs multiple times
  - Both phases use Diffie-Hellman key exchange to establish a shared key

# IKE Phase 1

---

- **Goal:** to establish a secure channel between two end points
  - This channel provides basic security features:
    - Source authentication
    - Data integrity and data confidentiality
    - Protection against replay attacks

# IKE Phase 1

---

- **Rationale:** each application has different security requirements
- But they all need to negotiate policies and exchange keys!
- So, provide the basic security features and allow application to establish custom sessions

## Examples

---

- Packets sent to **BofA.com** must be encrypted using AES with HMAC-SHA1 integrity check
- All packets sent to address **cnn.com** must be integrity checked with HMAC-SHA1 and no confidentiality is required.

# Phase 1 Exchange

---

- Can operate in two modes:
  - **Main mode**
    - Six messages in three round trips
    - More options
  - **Quick mode**
    - Four messages in two round trips
    - Less options

# Phase 1 (Main Mode)

---



Establish vocabulary for further communication

## Phase 1 (Main Mode) (2)

---

Initiator

Responder

Header,  $SA_1$



[Header,  $SA_1$ ]

[Header, KE,  $N_i$  { , Cert\_Req} ]



Establish secret key using Diffie-Hellman key exchange  
Use nonces to prevent replay attacks

# Phase 1 (Main Mode) (3)

---

Initiator

Responder

[Header, SA<sub>1</sub>]



[Header, SA<sub>1</sub>]

[Header, KE, Ni {, Cert\_req}]



[Header, KE, Nr {, Cert\_req}]

[Header, IDi, {CERT} sig]



[Header, IDr, {CERT} sig]

Signed hash of IDi (without Cert\_req , just send the hash)



# IPSec (Phase 1)

---

- Four different ways to authenticate (either mode)
  - Digital signature
  - Two forms of authentication with public key encryption
  - Pre-shared key
- **NOTE:** IKE does use public-key based cryptography for encryption

## IPSec (Phase 2)

---

- **Goal:** to establish custom secure channels between two end points
  - End points are identified by <IP, port>:
    - e.g. <128.9.70.63, 8000>
  - Or by destination network:
    - e.g. All packets going to 128.124.100.0/24
  - Use the secure channel established in Phase 1 for communication

## IPSec (Phase 2)

---

- **Only one mode:** Quick Mode
- Multiple quick mode exchanges can be multiplexed
- Generate SAs for two end points
- Can use secure channel established in phase 1

---

# **CSci530: Computer Security Systems**

## **Lecture 10 – 4 November 2022**

### **IPSec and Networking (cont)**

### **then Intrusion Detection**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

# IPsec Policy

---

- Phase 1 policies are defined in terms of *protection suites*
- Each protection suite
  - Must contain the following:
    - Encryption algorithm
    - Hash algorithm
    - Authentication method
    - Diffie-Hellman Group
  - May optionally contain the following:
    - Lifetime
    - ...

# IPSec Policy

---

- Phase 2 policies are defined in terms of *proposals*
- Each proposal:
  - May contain one or more of the following
    - AH sub-proposals
    - ESP sub-proposals
    - IPComp sub-proposals
    - Along with necessary attributes such as
      - Key length, life time, etc

# IPSec Policy Example

---

- In English:
  - All traffic to 128.104.120.0/24 must be:
    - Use pre-hashed key authentication
    - DH group is MODP with 1024-bit modulus
    - Hash algorithm is HMAC-SHA (128 bit key)
    - Encryption using 3DES
- In IPSec:
  - [Auth=Pre-Hash;  
DH=MODP(1024-bit);  
HASH=HMAC-SHA;  
ENC=3DES]

# IPsec Policy Example

---

- In English:
  - All traffic to 128.104.120.0/24 must use one of the following:
    - AH with HMAC-SHA or,
    - ESP with 3DES as encryption algorithm and (HMAC-MD5 or HMAC-SHA as hashing algorithm)
- In IPsec:
  - [AH: HMAC-SHA] or,
  - [ESP: (3DES and HMAC-MD5) or (3DES and HMAC-SHA)]



# Attack Paths

---

- **Many attacks today are staged from compromised machines.**
  - **Consider what this means for network perimeters, firewalls, and VPN's.**
- **A host connected to your network via a VPN is an unsecured perimeter**
  - **So, you must manage the endpoint even if it is your employees home machine.**

# Defense in Depth

---

- **One should apply multiple firewalls at different parts of a system.**
  - **These should be of different types.**
- **Consider also end to end approaches**
  - **Data architecture**
  - **Encryption**
  - **Authentication**
  - **Intrusion detection and response**

# **Case Study: Trusted Network Interpretation**

---

- **Focusing on the network structure supporting MLS policies with high assurance:**
- **[https://wiki.umn.edu/pub/CBI\\_ComputerSecurity/PubRainbowSeries/ncsc-tg-005red.pdf](https://wiki.umn.edu/pub/CBI_ComputerSecurity/PubRainbowSeries/ncsc-tg-005red.pdf)**
  - **MS CyberSecEng Students will want to Read chapter 1**

## **MLS components**

---

- **MLS (separating users of different clearance levels as they access different classification levels)**
  - **Clearance level**
  - **Classification level**
  - **Security level (generic term for either clearance level or classification level)**

# How to Support BLP

---

- **Separate Machines – System High**
- **Trusted High Assurance Systems**
  - **Capable of implementing MAC (e.g. BLP)**
- **What about networks?**

# Organization of the TNI

---

- Two parts, three appendices, a list of acronyms, a glossary, and a list of references.
- Part I presents TCSEC statements and detailed interpretations, which together constitute the requirements against which networks will be evaluated; and rationale for the network interpretation of the TCSEC. The TCSEC statement applies as modified by the Interpretation.
- Part II is a description of other Security Services not covered in the TCSEC interpretation which may be applicable to networks.
- Appendix A describes the evaluation of network components.
- Appendix B describes the rationale for network partitioning into individual components.

# **Network Interpretation of TCB**

---

- Like a stand-alone system, the network as a whole possesses a single TCB, referred to as the NTCB, consisting of the totality of security-relevant portions of the network.
- Unlike a stand-alone system, design and evaluation of the network rests on understanding how security mechanisms are distributed and allocated to various components, in such a way that the security policy is supported reliably in spite of (1) the vulnerability of the communication paths and (2) the concurrent, asynchronous operation of the network components.

# NTCB Partitions

---

- An NTCB that is distributed over a number of network components is referred to as partitioned, and that part of the NTCB residing in a given component is referred to as an NTCB partition. [TNI]pg13



# Background

---

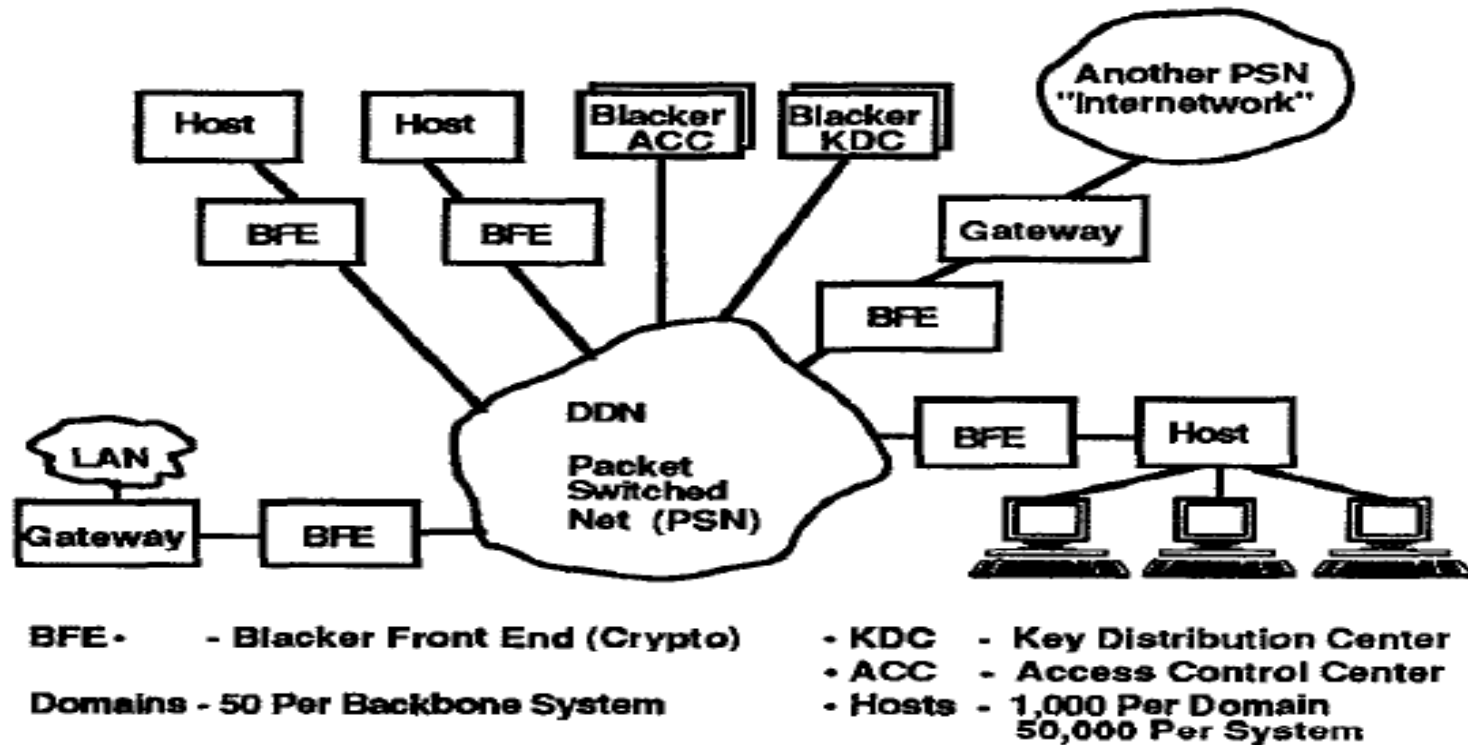
- **Some Security Policy Enforcement Tools & Components**
  - **Blacker**
  - **IPSEC**
  - **IPSO/CIPSO security options**
  - **HAIPE**
  - **Suite B**
  - **Encryption & Authentication at various protocol layers**
  - **Firewalls**
  - **Proxy servers**

# Evolution of Network MAC Enforcement

---

- Link encryptors → network encryptors
  - Internet Private Line Interface (IPLI) ~1983
    - (experimental hardware similar to IPSEC/HAIPE devices)
  - Blacker ~1984-1990
    - Provided for interconnection of MLS systems
    - Fielded production units but not available today
  - IP Security Options (IPSO) ~1988 (term re-used)
  - Trunk Encryptors ~1993
    - Passed “telecom signalling” through the encryptor
  - ATM Encryptors ~1994-1998
    - First “key agile” encryptors – different key for each ATM cell stream
  - IPSEC ~1998
  - HAIPE ~2001
    - Provided for interconnection of Single Level Systems
  - MPLS – Multi-Protocol Label Switching
  - VLANs

# Blacker



**Figure 1. BLACKER-DDN Architecture**

## **IPSO (RFC 1108)**

---

- **Table 1. Classification Level Encodings (each encoding is hamming distance 4 from the others)**

<b>Value</b>	<b>Name</b>
– 00000001	– (Reserved 4)
– 00111101	– Top Secret
– 01011010	– Secret
– 10010110	– Confidential
– 01100110	– (Reserved 3)
– 11001100	– (Reserved 2)
– 10101011	– Unclassified
– 11110001	– (Reserved 1)

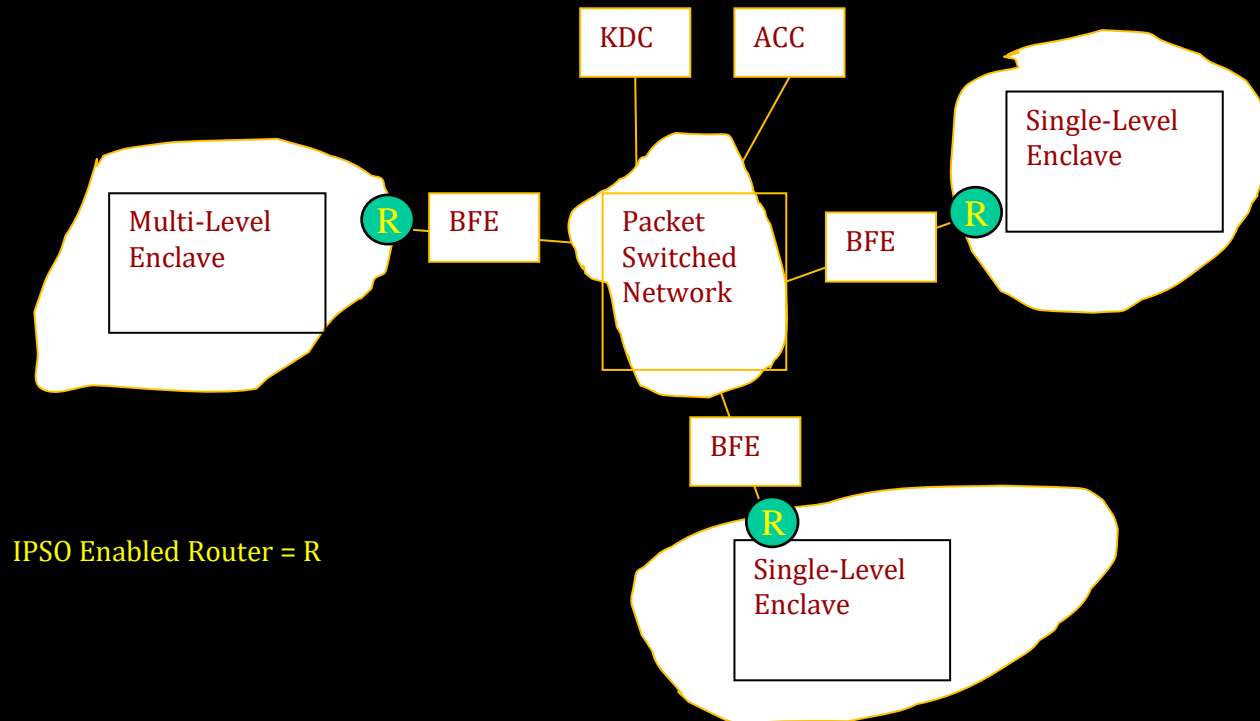
## Protection Authority Field

---

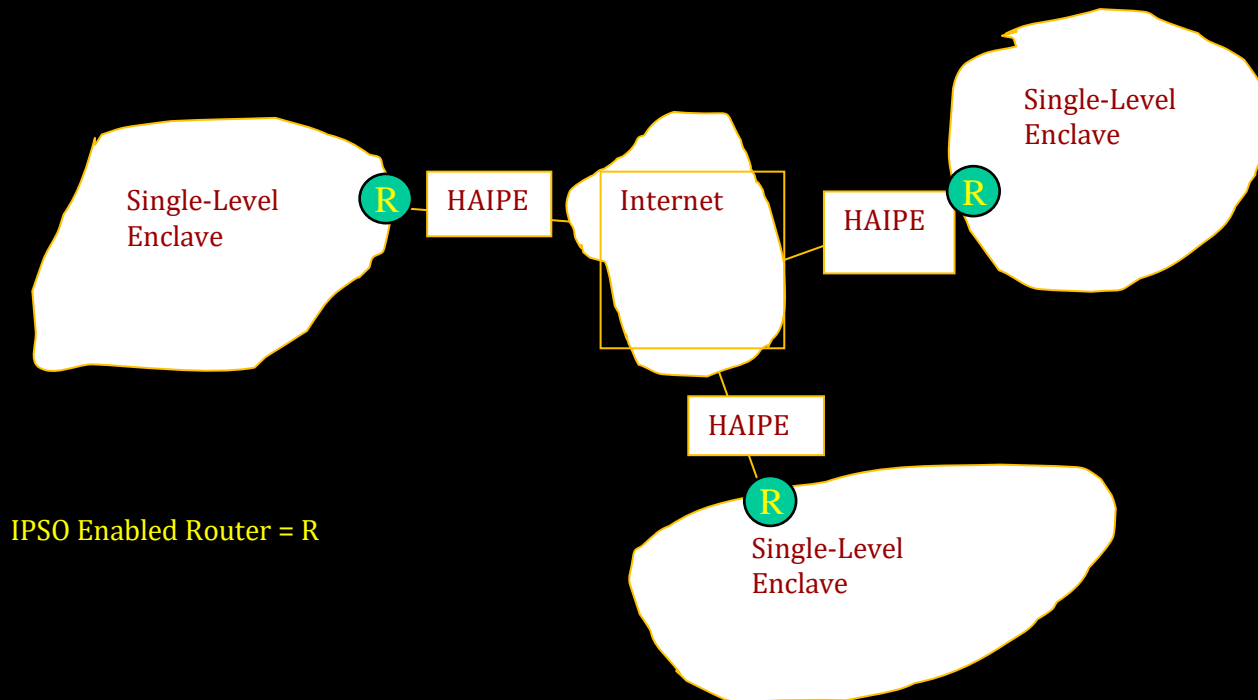
### • Table 2 - Protection Authority Bit Assignments

BIT NUMBER	AUTHORITY
– 0	<b>GENSER</b>
– 1	<b>SIOP-ESI</b>
– 2	<b>SCI</b>
– 3	<b>NSA</b>
– 4	<b>DOE</b>
– 5, 6	<b>Unassigned</b>
– 7	<b>Field Termination Indicator</b>

# IPSO for Labeling



# IPSO in Single Level Networks



## Summary

---

- IPSO provides a way to add sensitivity labels to datagrams
- IPSEC (/HAIP) provides a way to protect the integrity and confidentiality of these datagrams between
  - Single level networks
  - Multi-level networks (requires MLS computing nodes to complete the protection within the interconnected networks)



## Summary (2)

---

- **Blacker, IPSEC/HAIPE, IPSO are attempts to insert MAC enforcement at the boundaries between network enclaves**
- **Firewalls, bastion hosts, proxy servers, etc. are attempts to insert Application level policy enforcement at the boundaries between network enclaves**

# Protecting the Inside

---

- **Firewalls are better at protecting inward threats.**
  - **But they can prevent connections to restricted outside locations.**
  - **Application proxies can do filtering for allowed outside destinations.**
  - **Still need to protect against malicious code.**
- **Standalone (i.e. not host based) firewalls provide stronger self protection.**

---

# **CSci530: Computer Security Systems**

## **Lecture 10 – 4 November 2022**

### **Intrusion Detection**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

# Intrusion Types

---

- **External attacks**
  - Password cracks, port scans, packet spoofing, DOS attacks
- **Internal attacks**
  - Masqueraders, Misuse of privileges

# Attack Stages

---

- **Intelligence gathering**
  - attacker observes the system to determine vulnerabilities (e.g, port scans)
- **Planning**
  - decide what resource to attack and how
- **Attack execution**
  - carry out the plan
- **Hiding**
  - cover traces of attack
- **Preparation for future attacks**
  - install backdoors for future entry points

# **Intrusion Detection**

---

- **Intrusion detection is the problem of identifying unauthorized use, misuse, and abuse of computer systems by both system insiders and external penetrators**
- **Why Is IDS Necessary?**

# IDS types

---

- **Detection Method**
  - Knowledge-based (signature-based ) vs behavior-based (anomaly-based)
- **Behavior on detection**
  - passive vs. reactive
- **Deployment**
  - network-based, host-based and application -based

# **Components of ID systems**

---

- **Collectors**
  - **Gather raw data**
- **Director**
  - **Reduces incoming traffic and finds relationships**
- **Notifier**
  - **Accepts data from director and takes appropriate action**



# **Advanced IDS models**

---

- **Distributed Detection**
  - **Combining host and network monitoring (DIDS)**
  - **Autonomous agents (Crosbie and Spafford)**

# **Intrusion Response**

---

- **Intrusion Prevention**
  - (marketing buzzword)
- **Intrusion Response**
  - How to react when an intrusion is detected

## Possible Responses

---

- Notify administrator
- System or network lockdown
- Place attacker in controlled environment
- Slow the system for offending processes
- Kill the process

## **Phase of Response** (Bishop)

---

- Preparation
- Identification
- Containment
- Eradication
- Recovery
- Follow up

# **PREPARATION**

---

- **Generate baseline for system**
  - **Checksums of binaries**
    - **For use by systems like tripwire**
- **Develop procedures to follow**
- **Maintain backups**

# IDENTIFICATION

---

- This is the role of the ID system
  - Detect attack
  - Characterize attack
  - Try to assess motives of attack
  - Determine what has been affected

# CONTAINMENT

---

- **Passive monitoring**
  - To learn intent of attacker
  - Learn new attack modes so one can defend against them later
- **Constraining access**
  - Locking down system
  - Closing connections
  - Blocking at firewall, or closer to source
- **Combination**
  - Constrain activities, but don't let attacker know one is doing so (Honeypots, Jail).

# ERADICATION

---

- Prevent attack or effects of attack from recurring.
  - Locking down system (also in containment phase)
  - Blocking connections at firewall
  - Isolate potential targets



# RECOVERY

---

- Restore system to safe state
  - Check all software for backdoors
  - Recover data from backup
  - Reinstall but don't get re-infected before patches applied.

# **FOLLOWUP**

---

- **Take action against attacker.**
  - **Find origin of attack**
- **Notify other affected parties**
  - **Some of this occurs in earlier phases as well**
- **Assess what went wrong and correct procedures.**
- **Find buggy software that was exploited and fix**

# Limitations of Monolithic ID

---

- Single point of failure
- Limited access to data sources
- Only one perspective on transactions
- Some attacks are inherently distributed
  - Smurf
  - DDoS
- Conclusion: “Complete solutions” aren’t

# Sharing Information

---

- **Benefits**
  - Increased robustness
  - More information for all components
  - Broader perspective on attacks
  - Capture distributed attacks
- **Risks**
  - Eavesdroppers, compromised components
  - In part – resolved cryptographically

# Sharing Intrusion Information

---

- Defining appropriate level of expression
  - Efficiency
  - Expressivity
  - Specificity

# **CIDF**

---

- **Common Intrusion Detection Framework**
  - Collaborative work of DARPA-funded projects in late 1990s
  - Task: Define language, protocols to exchange information about attacks and responses

# CISL

---

- **Common Intrusion Specification Language**
  - **Conveys information about attacks using ordinary English words**
  - **E.g., User joe obtains root access on demon.example.com at 2003 Jun 12 14:15 PDT**

# CISL

---

- Problem: Parsing English is hard
- S-expressions (Rivest)
  - Lisp-like grouping using parentheses
  - Simplest examples: (name value) pairs  
(Username 'joe')  
(Hostname 'demon.example.com')  
(Date '2003 Jun 12 14:15 PDT')  
(Action obtainRootAccess)



# CISL

---

- **Problems with simple pairs**
  - **Confusion about roles played by entities**
    - **Is joe an attacker, an observer, or a victim?**
    - **Is demon.example.com the source or the target of the attack?**
  - **Inability to express compound events**
    - **Can't distinguish attackers in multiple stages**
- **Group objects into GIDOs**

# CISL: Roles

---

- Clarifies roles identified by descriptors  
    (Attacker  
        (Username 'joe')  
        (Hostname 'carton.example.com')  
        (UserID 501)  
    )  
    (Target  
        (Hostname 'demon.example.com')  
    )

# CISL: Verbs

---

- Permit generic description of actions  
(Compromise  
  (Attacker ...)  
  (Observer  
    (Date '2003 Jun 12 14:15 PDT')  
    (ProgramName 'GrIDSDetector')  
  )  
  (Target ...)  
)

# Lessons from CISL

---

- Lessons from testing, standardization efforts
  - Heavyweight
  - Not ambiguous, but too many ways to say the same thing
  - Mismatch between what CISL can say and what detectors/analyzers can *reliably* know

# **Worm and DDOS Detection**

---

- **Difficulty is distinguishing attacks from the background.**
  - **Zero Day Worms**
  - **DDoS**
- **Discussion of techniques**
  - **Honeynets, network telescopes**
  - **Look for correlation of activity**

# Reacting to Attacks

---

- **How to Respond to Ongoing Attack**
  - Disable attacks in one's own space
  - Possibly observe activities
  - Beware of rules that protect the privacy of the attacker (yes, really)
  - Document, and establish chain of custody.
- **Do not retaliate**
  - May be wrong about source of attack.
  - May cause more harm than attack itself.
  - Creates new way to mount attack
    - Exploits the human element

# Virus Checking

---

- **Signature based**
  - Looks for known indicators in files
  - Real-time checking causes files to be scanned as they are brought over to computer (web pages, email messages) or before execution.
  - On server and client
- **Activity based**
  - Related to firewalls, if look for communication
  - Alert before writing to boot sector, etc.
- **Defenses beyond just checking**
  - Don't run as root or admin

---

# **CSci530: Security Systems**

## **Lecture 11 – November 12, 2021**

### **The Human Element**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**



# **The Human is the Weak Point**

---

- **Low bandwidth used between computer and human.**
  - **User can read, but unable to process crypto in head.**
  - **Needs system as its proxy**
  - **This creates vulnerability.**
- **Users don't understand system**
  - **Often trust what is displayed**
  - **Basis for phishing**

# **The Human is the Weak Point(2)**

---

- **Humans make mistakes**
  - **Configure system incorrectly**
- **Humans can be compromised**
  - **Bribes**
  - **Social Engineering**
- **Programmers often don't consider the limitations of users when designing systems.**

# Some Attacks

---

- **Social Engineering**
  - **Phishing – in many forms**
- **Mis-configuration**
- **Carelessness**
- **Malicious insiders**
- **Bugs in software**

# Addressing the Limitations

---

- **Personal Proxies**
  - Smartcards or devices
- **User interface improvements**
  - Software can highlight things that it thinks are odd.
- **Delegate management**
  - Users can rely on better trained entities to manage their systems.
- **Try not to get in the way of the users legitimate activities**
  - Or they will disable security mechanisms.

# **Social Engineering**

---

- **Arun Viswanathan provided me with some slides on social engineering that we wrote based on the book “The Art of Deception” by Kevin Mitnik.**
  - **In the next 6 slides, I present material provided by Arun.**
- **Social Engineering attacks rely on human tendency to trust, fooling users that might otherwise follow good practices to do things that they would not otherwise do.**

# **Total Security / not quite**

---

- **Consider the statement that the only secure computer is one that is turned off and/or disconnected from the network.**
- **The social engineering attack against such systems is to convince someone to turn it on and plug it back into the network.**

# **Six Tendencies**

---

- **Robert B. Cialdini summarized six tendencies of human nature in the February 2001 issue of Scientific American.**
- **These tendencies are used in social engineering to obtain assistance from unsuspecting employees.**

# Six Tendencies

---

- People tend to comply with requests from those in authority.
  - Claims by attacker that they are from the IT department or the audit department.
- People tend to comply with request from those who they like.
  - Attackers learns interests of employee and strikes up a discussion.



# Six Tendencies

---

- People tend to follow requests if they get something of value.
  - Subject asked to install software to get a free gift.
- People tend to follow requests to abide by public commitments.
  - Asked to abide by security policy & demonstrate compliance by disclosing their password is secure:
    - Were talking about cybersecurity today and how safe peoples passwords are, what is your online password?

# Six Tendencies

---

- **People tend to follow group norms.**
  - **Attacker mentions names of others who have “complied” with the request, and will the subject comply as well.**
- **People tend to follow requests under time commitment.**
  - **First 10 callers get some benefit.**

# Steps of Social Engineering

---

- **Conduct research**
  - Get information from public records, company phone books, company web site, checking the trash.
- **Developing rapport with subject**
  - Use information from research phase. Cite common acquaintances, why the subjects help is important.
- **Exploiting trust**
  - Asking subject to take an action. Manipulate subject to contact attacker (e.g. phishing).
- **Utilize information obtained from attack**
  - Repeating the cycle.

# Phishing

---

- **A website (or other form of interaction) where the user believes they are communicating with an entity they trust but are actually communicating with the attacker.**
    - **Could be a phone call claiming to be from your bank.**
    - **Could be a paper letter that appears genuine.**
      - **Most commonly it is a link in a email message**
      - **Or a search result**
      - **Or a link on a web page**
      - **Mistyped domain name (typosquatting)**
- Visible text of link might show name or even URL of legitimate site, but target of link is different**
- Sometimes subtly different**
- letter or character or prefix**

# Other Redirections

---

- **Man in the Middle**
  - **At free hotspots**
  - **Through hacks such as superfish**
- **Domain name hijacking**
  - **Simple malware**
  - **Cache poisoning**
    - **(in a couple of slides)**

# **Context Sensitive Certificate Verification and Specific Password Warnings**

---

- **Work out of University of Pittsburgh**
- **Changes dialogue for accepting signatures by unknown CAs.**
- **Changes dialogue to prompt user about situation where password are sent unprotected.**
- **Does reduce man in the middle attacks and phishing**
  - **By preventing easy acceptance of CA certs**
  - **Requires specific action to retrieve cert**
  - **Would users find a way around this?**

---

# **CSci530: Computer Security Systems**

## **Lecture 12 – 18 November 2022**

### **The Domain Name System**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

# **The Domain Name System Overview**

---

- **Overview of the Domain Name System**
  - **Recursive and Iterative Queries**
  - **Recursive or Caching DNS Server**
  - **Extreme resilience to DoS attacks**
- **The DNS Protocol**
  - **What is in a DNS record and response**
  - **Reverse DNS lookup**



# What is the Domain Name System

---

- An infrastructure for mapping a name to and IP address (and to other data too).
  - A name is a string like `www.google.com` used by humans to identify a system
  - An address is an IP address like `128.9.128.127` used within the network to identify a computer and to help determine a route.
- The DNS is an application-layer protocol used by hosts to query DNS servers and the distributed database implemented by a hierarchy of name servers
  - It is described in RFC 1034 and RFC 1035
- It supports Iterative and Recursive queries with caching
- It is deployed to support Reverse DNS lookup

## **Why Cover this in a Class on Security**

---

- **Naming and security go hand in hand**
- **Many classic attacks are facilitated by attacks in name resolution.**
- **Many security problems stem from incorrect assumptions about name resolution.**
- **DNSSec (DNS Security) implement its its own PKI**
  - **It is still important to understand its limitation**
  - **But it provides a good case study**

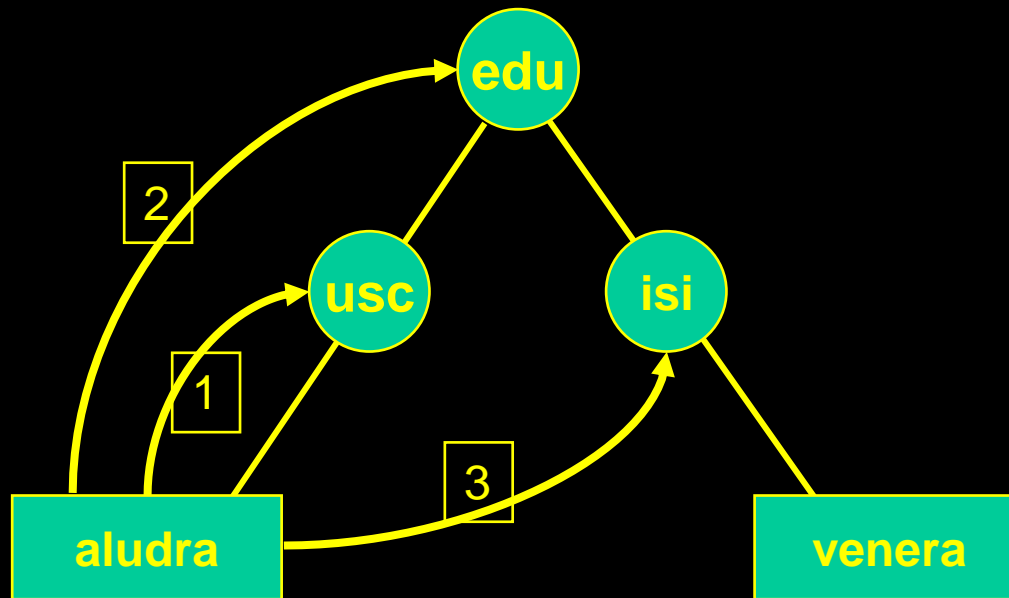
# **The Domain Name System Evolution**

---

- **Historically the host file was located on the local computer**
  - **E.g., c:\windows\system32\drivers\etc\hosts**
  - **Need to be maintained and updated by an administrator**
- **Maintaining the hosts files for all Internet domain names and sub domains is not feasible**
  - **So distributed database (DNS) was developed at ISI**
  - **Service is run by many organizations, Overseen by ICANN, with top level servers across the world. Several at USC and ISI.**
  - **Facilitates the mapping of domains (and URL's) to IP addresses.**

# Domain Name System

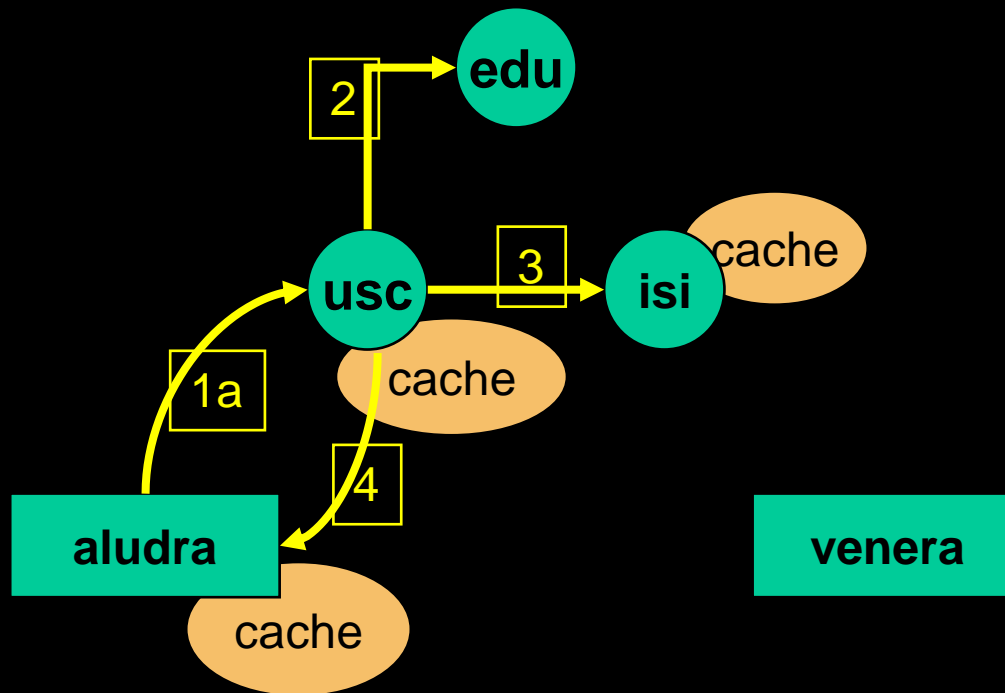
Iterative query



Lookup(venera.isi.edu)

# Caching in the Domain Name System

---



# DNS Root Name Servers

---

- Contacted by local DNS name server that can not resolve a name
- A local DNS system is pre-configured with the known addresses of the root servers in a file using root hints
  - This file must be updated periodically by the local administrator
- Root name servers:
  - The root name servers know which servers are responsible for the top-level domains (TLD), such as .edu
  - Each top-level domain (such as .edu) has its own set of servers
    - TLD servers in turn delegate to the name servers responsible for individual domain names (such as ns.mit.edu)
  - Two of the root servers are managed by USC
- 13 organizations manage the root DNS servers
- The locations of the root servers [www.root-servers.org](http://www.root-servers.org)
- (rewrite)

# Authoritative Name Servers

---

- Each ISP, company, university, organization has at least one default name server
- When host makes a DNS query,
  - Query is sent to its local authoritative DNS server or a recursive server
  - DNS server acts as proxy, forwards query into the DNS hierarchy: recursive query
- The DNS information for one domain name is stored as resource record(s) (RR's)
- A DNS zone is a portion of the global Domain Name System (DNS) namespace for which administrative responsibility has been delegated (rewrite)

# DNS Resolver

---

- Inside a host, a process called DNS resolver obtains the mapping from name to IP address
  - RESOLVERS are programs that obtain information from name servers in response to client requests
  - A cache preserves a mapping for certain amount of time
  - A DNS resolver can be running inside a computer that is
    - A client computer
    - A web server, mail server, etc.
    - A DNS server
- Resolvers must have access to at least one name server
  - Use that name server's information to answer a query directly
  - Perform the query using referrals to other name servers
  - (rewrite)



# DNS resource record (RR) Type

---

RR format: (name, [pref.], value, type, [ttl])

- **Type=A**
  - Name is host's name
  - Value is IP address
- **Type=NS**
  - Name is domain name (e.g. auburn.edu)
  - Value is name of authoritative name server for this domain (e.g. dns.auburn.edu)
- **Type=MX**
  - Name is domain name (e.g. auburn.edu)
  - Value is name of mail server designed for the domain (e.g. aumail.duc.auburn.edu)
  - A preference value is designated for each mail server if there are multiple MX RR's in a domain

# DNS resource record (RR) Type

---

- **Type=CNAME**
  - Name (such as `www.ibm.com`) is alias name for “canonical” (real) name
  - Value is canonical name (such as `servereast.backup2.ibm.com`)
  - `www.ibm.com` (name) is really `servereast.backup2.ibm.com` (value)
- **Type=AAAA**
  - IPv6 host address (AAAA) resource record
  - Maps a DNS domain name to an Internet Protocol (IP) version 6 128-bit address
- **TTL: time to live in cache**
  - 32 bit integer for the number of seconds

# Queries to the dns

---

## Common tools:

- **Nslookup**
- **Dig**

# DNS Caching

---

- DNS responses are cached
  - Quick response for repeated translations
- DNS negative queries are also cached
  - For example, misspellings
- Cached data periodically times out
- Cache poisoning for pharming
  - Redirect website's traffic to bogus website by forging DNS mapping
  - An attacker attempts to insert a fake address record for an Internet domain into the DNS
    - If the server accepts the fake record, the cache is poisoned and subsequent requests for the address of the domain are answered with the address of a server controlled by the attacker
    - For as long as the fake entry is cached by the server (entries usually have a time to live (TTL) of a couple of hours) subscriber's browsers or e-mail servers will automatically go to the address provided by the compromised DNS server

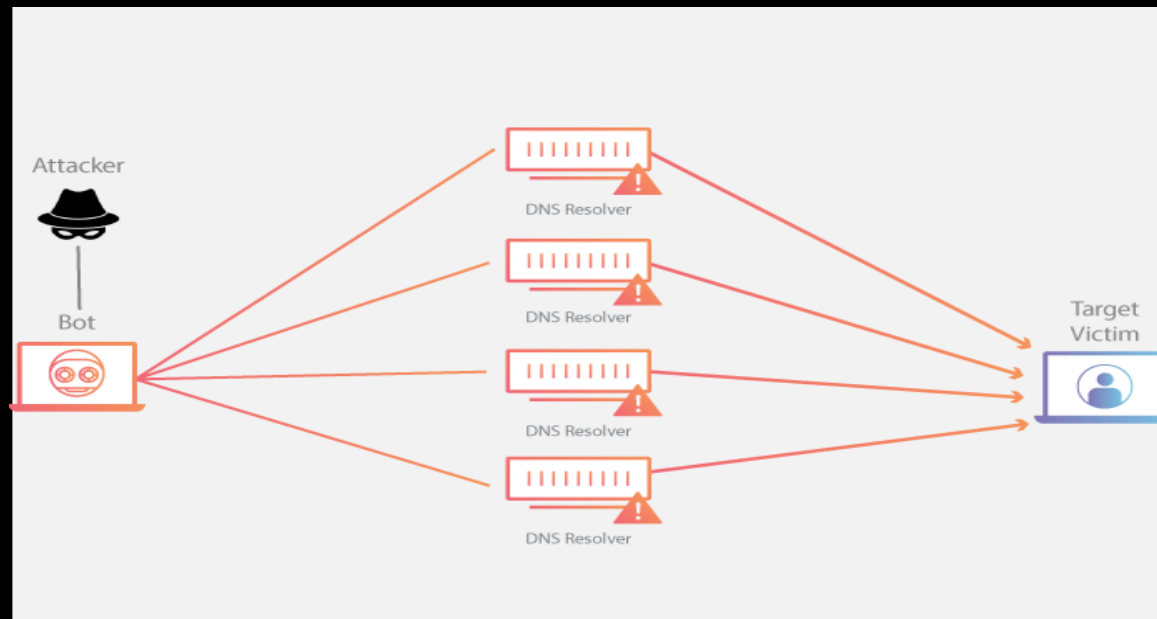
# DNS Amplification Attacks

Vishal Srinivasan

# What are DNS Amplification attacks?

- Uses open DNS resolvers to launch a denial of service attack
- Open DNS resolvers are those which resolve DNS queries for anyone on the internet
- The 'amplification' implies a multiplicative factor on the amount of data actually transmitted by the attacker to the amount which hits the target
- Primarily uses DNS 'ANY' query type to amplify the response traffic
- Can be exacerbated by the use of crafted domains with large DNS records and DNSSEC

# What are DNS Amplification attacks?



Source: cloudflare.com

# How much amplification can you achieve?

- Classic DNS protocol limits responses to 512 bytes
- Typical DNS requests are 20-60 bytes
- Therefore amplification of 12.8 is achievable
  
- With DNSSEC and 'ANY' queries you can have responses of 4096 bytes
- This raises the amplification to over 100



# How much amplification can you achieve?

- Cloudflare experienced almost 2Tbps DDoS attack last week
- The attack lasted for 1min and included DNS amplification with a combination of UDP floods
- Cloudflare identified this attack to originate from over 15,000 bots running the Mirai botnet

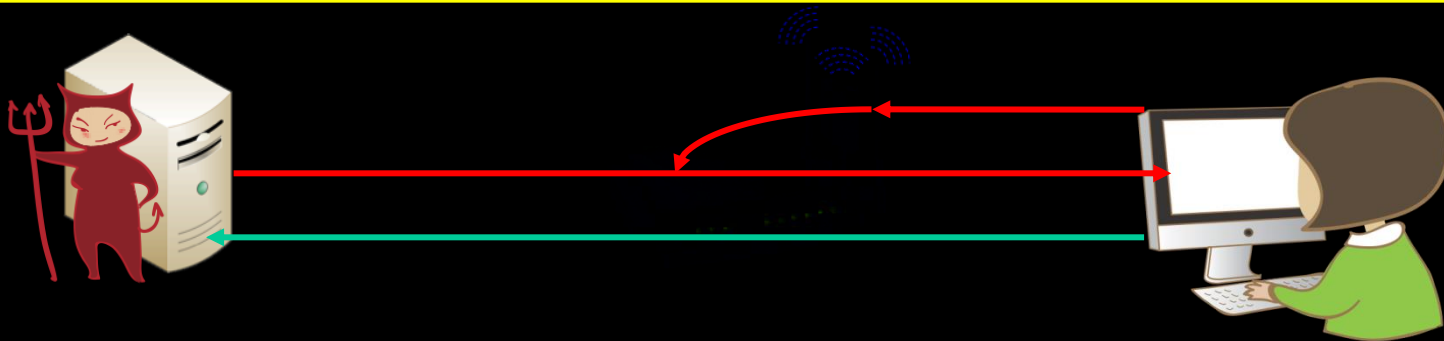
# How do we mitigate these attacks?

- Prevent IP spoofing using ingress filtering
  - Requires cooperation between ISPs globally
- Restrict or block 'ANY' queries
- Reduce the number of open DNS resolvers and configure them correctly
- Employ DDoS mitigation services such as Cloudflare and others

# References

1. R. van Rijswijk-Deij, A. Sperotto, and A. Pras, “DNSSEC and its potential for DDoS attacks,” Proceedings of the 2014 Conference on Internet Measurement Conference. ACM, Nov. 05, 2014. doi: 10.1145/2663716.2663731.
2. <https://www.cloudflare.com/learning/ddos/dns-amplification-ddos-attack/>
3. <https://blog.cloudflare.com/cloudflare-blocks-an-almost-2-tbps-multi-vector-ddos-attack/>

# Drive-By Pharming



- Alice is visiting a malicious site
- Malicious scripts is loaded to Alice's computer
- Malicious scripts discover router
- Crack the password of the router and login
  - Most home routers have default password
- Modify DNS setting in the router to a name server controlled by attacker
  - Alice will be visiting bogus sites since DNS provides mappings to sites forged by attacker
  - Capture critical information by bogus sites

# DNS Vulnerabilities

---

- **Deployed DNS may include no authentication**
  - Any DNS response is generally believed
  - No validating mechanism for the authenticity of information
- **When a DNS caching server gets a query from a subscriber for a domain, it looks to see if it has an entry cached**
  - If it does not, it asks authoritative DNS servers (run by domain owners) and waits for their responses
  - First response wins the cache acceptance

# **DNS cache poisoning**

---

- **Prior to Dan Kaminsky's discovery in 2008, attackers could only exploit this narrow opening**
  - **They had to beat legitimate authoritative DNS servers by sending a fake query response, hoping they arrive at the caching server first with the correct query parameter value**
    - **The same IP address it was sent from**
    - **The same port number it was sent from**
    - **The answer matches the question asked**
    - **A unique ID number matches what was sent**
  - **These races typically only lasted a fraction of a second, making it difficult for an attacker to succeed**

# More DNS cache poisoning

---

- Dan Kaminsky discovered this new vulnerability because a security researcher figured out a way to eliminate the narrow time window
- The ID that the attacker needs to guess is not fully random (or not random at all)
- Attacker rapidly firing questions at the caching server that an attacker knows the server will not be able to answer
  - E.g., an attacker can ask where x1y2z3.amazon.com is, knowing a caching server is unlikely to have such an entry
  - That provokes subsequent questions from the caching server and creates millions of opportunities to send fake answers by attacker

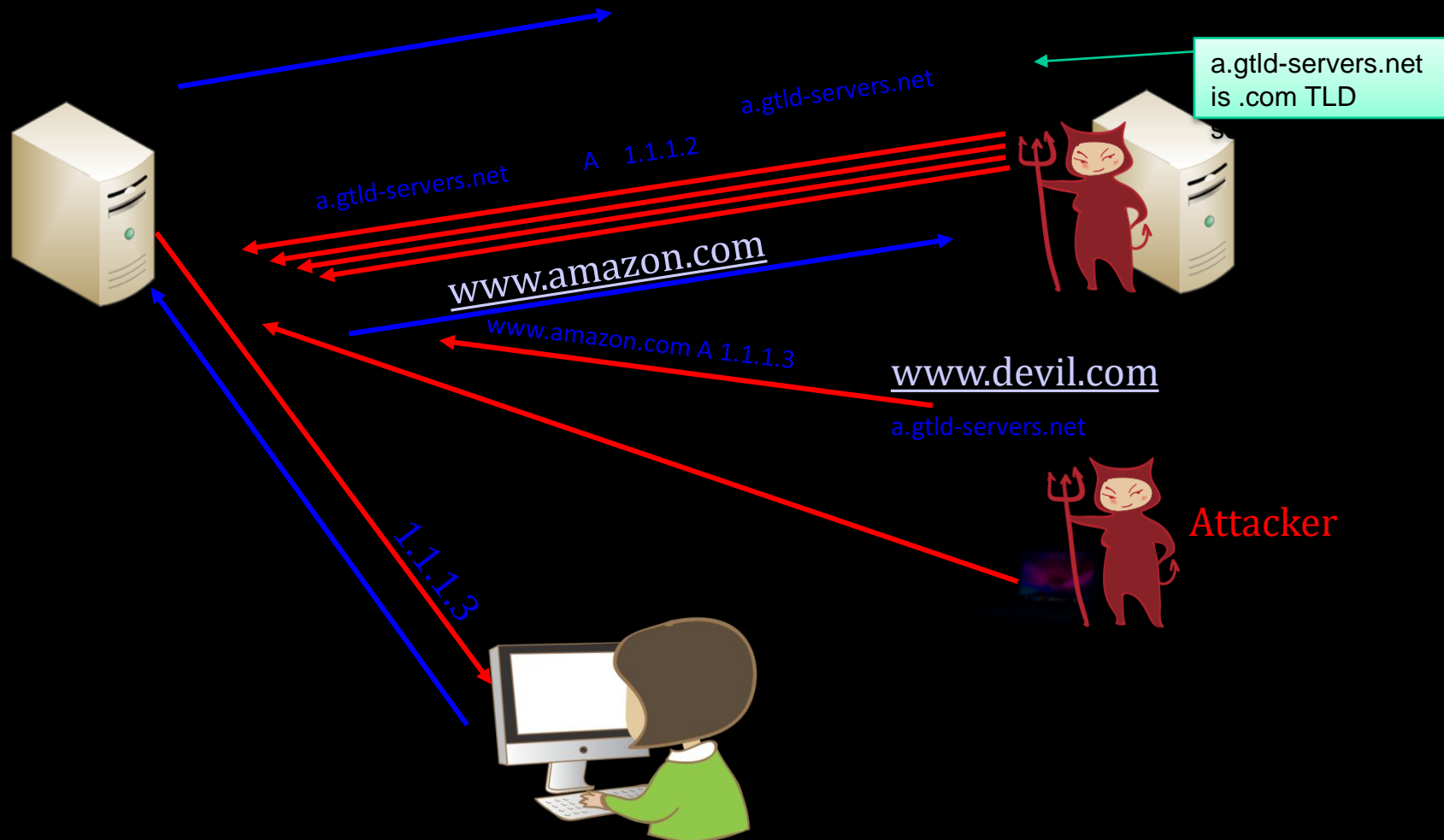
# DNS cache poisoning

---

- In the fake answers, the attacker also points the caching server to a fake name-server's IP address (1.1.1.2) for the domain, amazon.com
  - The additional section of the reply packet contains the bogus IP address, 1.1.1.2, for pdns1.amazon.com (the name of real amazon.com's DNS server)
- Every subsequent query for the domain, amazon.com, will be directed to the attacker's server at 1.1.1.2.
- This means the users at banka.com now are using bogus address mapping in the domain: amazon.com
- If a name server provides both recursive and authoritative name service, a successful attack on the recursive portion can store bad data that is given to computers that want authoritative answers
- it was demonstrated that open source DNS servers could be compromised in 10 seconds
- TLD DNS can be modified in cache too



# Cache Poisoning one TLD



# Short-term Defense

---

- The patches that have been released in 2008 randomize the source port for the recursive Server
  - UDP port used for a query should no longer be the default port 53, but rather a port randomly chosen from the entire range of UDP ports (not including the reserved ports)
  - Microsoft's updated DNS server is said to use 11 bits for randomizing about 2,500 UDP ports
- Makes it harder for an attacker to guess query parameters
  - Both the 16-bit query ID and as many as 11 additional bits for the UDP port must be correct, for a total of up to 134 million combinations
  - $2^{16} \cdot 2^{11} = 2^{27} = 1.34 \cdot 10^8$
- **DNS servers behind network address translation (NAT): most NATs de-randomized the UDP ports used by the DNS server, rendering the new fix less effective**
- Another security researcher demonstrated that it was still possible to poison a DNS server even with the protection afforded by randomization across 64,000 UDP ports

# **Long-term solution: authentication**

---

- Resolver can not distinguish between valid and invalid data in a response
- Idea is to add source authentication
  - Verify the data received in a response is the same as that entered by the zone administrator (ON AN END TO END BASIS)
- DNSSEC (DNS Security Extensions) protects against data spoofing and corruption
- DNSSEC also provides mechanisms to authenticate servers and requests
- DNSSEC provides mechanisms to establish authenticity and integrity

# Authenticating DNS Responses

---

- Each DNS zone signs its data using a private key
  - Recommend signing done offline in advance
- Query for a particular record returns:
  - The requested resource record set
  - A signature (RRSIG) of the requested resource record set (RRset)
- Resolver authenticates response using public key
  - Public key is pre-configured or learned via a sequence of key records in the DNS hierarchy

# DNS Security Extensions (1)

---

- **DNSSEC allows RR's and zones to have origin authentication and integrity**
  - **One private key signs one zone**
    - **Use this case as the example since it is simple to understand**
  - **It is possible to use multiple private keys for signing a zone**
- **The Zone Signing Key (ZSK) can be used to sign all the data in a zone on a regular basis**
  - **When a Zone Signing Key is to be rolled, no interaction with the parent is needed**
  - **This allows for signature validity periods on the order of days**
- **The Key Signing Key (KSK) is only to be used to sign the DNSKEY RRs, containing ZSK, in a zone**
  - **If a Key Signing Key is to be rolled over, there will be interactions with parties other than the zone administrator**

# DNS Security Extensions (2)

---

- **New types of RR's for DNSSEC**
  - **DNSKEY RR: Public key resource record**
    - Contains the public key
  - **RRSIG: Signature resource record**
    - Each RRset has its corresponding RRSIG
  - **DS: Delegation Signer (optional)**
    - A parent domain can optionally delegate to a new key pair for signing RR's in the child domain
    - Containing a digest
  - **NSEC: Next resource record**
    - Enables the DNS server to inform the client that a particular domain or type does not exist

# DNS Security Extensions (3)

---

- **DNSKEY: Public key resource record**
  - A zone signs its authoritative resource record sets (RRsets) by using a private key and stores the corresponding public key in a DNSKEY RR
  - A resolver can then use the public key to validate signatures covering the RRsets in the zone, and thus to authenticate them
- **RRSIG: Signature resource record**
  - Each RRset has its corresponding RRSIG, containing a public-key signature which is stored as a resource record
    - E.g., www.x.com RR (type A) has a RRSIG RR containing the signature
    - The algorithm used (RSA/SHA1) to create the signature is contained in the RRSIG
    - The valid period of the RRSIG is also contained in RRSIG
    - RRSIG's are computed for every RRset in a zone file and stored
    - Add the corresponding pre-calculated signature for each RRset in answers to queries

# DNS Security Extensions (3)

---

- **DS: Delegation Signer (optional)**
  - When the parent zone delegates the name resolution to a child zone, the private key for signing is usually changed
    - E.g., .com DNS server has a pair of keys for signing and verifying .com zone
    - x.com has its own key pair for signing and verifying x.com zone
    - www.x.com RR is signed by x.com's private key
  - Each DNSKEY of a zone has a corresponding DS RR
  - DS RR contains the digest of the corresponding DNSKEY
    - E.g., SHA-1 is the algorithm to generate the digest
  - RRset in the zone x.com is verified using public key in DNSKEY(x.com)
- **NSEC: Next resource record**
  - Enables the DNS server to inform the client that a particular domain or type does not exist



# NSEC RR (1)

---

- Provides authenticated denial of existence for DNS data
  - Providing negative responses with the same level of authentication and integrity
- Defeat the attack discovered by Kaminsky
- The NSEC record allows a resolver to authenticate a negative reply for either name or type non-existence with the same mechanisms used to authenticate other DNS replies
- NSEC3 RR
  - Format and use the same as the NSEC Record
  - Uses hashed names instead of cleartext
- Use of NSEC records requires a canonical representation and ordering for domain names in zones
  - Chains of NSEC records explicitly describe the gaps, or "empty space", between domain names in a zone and list the types of RRsets present at existing names

# NSEC RR (4)

---

- The pseudo format (containing only the important fields) of NSEC RRs covering the gaps in the namespace relating to domain names and RR types found at each name

```
x.com. IN NSEC mail.x.com. (NS SOA MX RRSIG NSEC)
      IN RRSIG ( NSEC )
mail.x.com. IN NSEC ns.x.com. (A RRSIG NSEC)
           IN RRSIG ( NSEC )
ns.x.com. IN NSEC p.x.com. (A RRSIG NSEC)
          IN RRSIG ( NSEC )
p.x.com. IN NSEC s.x.com. (A RRSIG NSEC)
         IN RRSIG ( NSEC )
s.x.com. IN NSEC www.x.com. (NS RRSIG NSEC)
        IN RRSIG ( NSEC )
www.x.com. IN NSEC x.com. (A RRSIG NSEC)
           IN RRSIG ( NSEC )
```

# PKI: chain of trust (1)

---

- By using the hierarchical property of the DNS, DNSSEC can verify signatures without configuring the public keys of every single domain
- PKI allows a DNS cache server/resolver to verify signatures by tracing from a trusted anchor's key down the DNS delegation chain
- Each level of the DNS must deploy DNSSEC
- Resolver can learn a zone's public key by having a trust anchor configured into the resolver
  - Trusted anchor:
    - Forming an authentication chain from a newly learned public key back to a previously known authenticated public key, which in turn either has been configured into the resolver or must have been learned and verified previously
    - Therefore, a resolver must be configured with at least one trust anchor's public key initially
      - The KSK of the root server published by ICANN

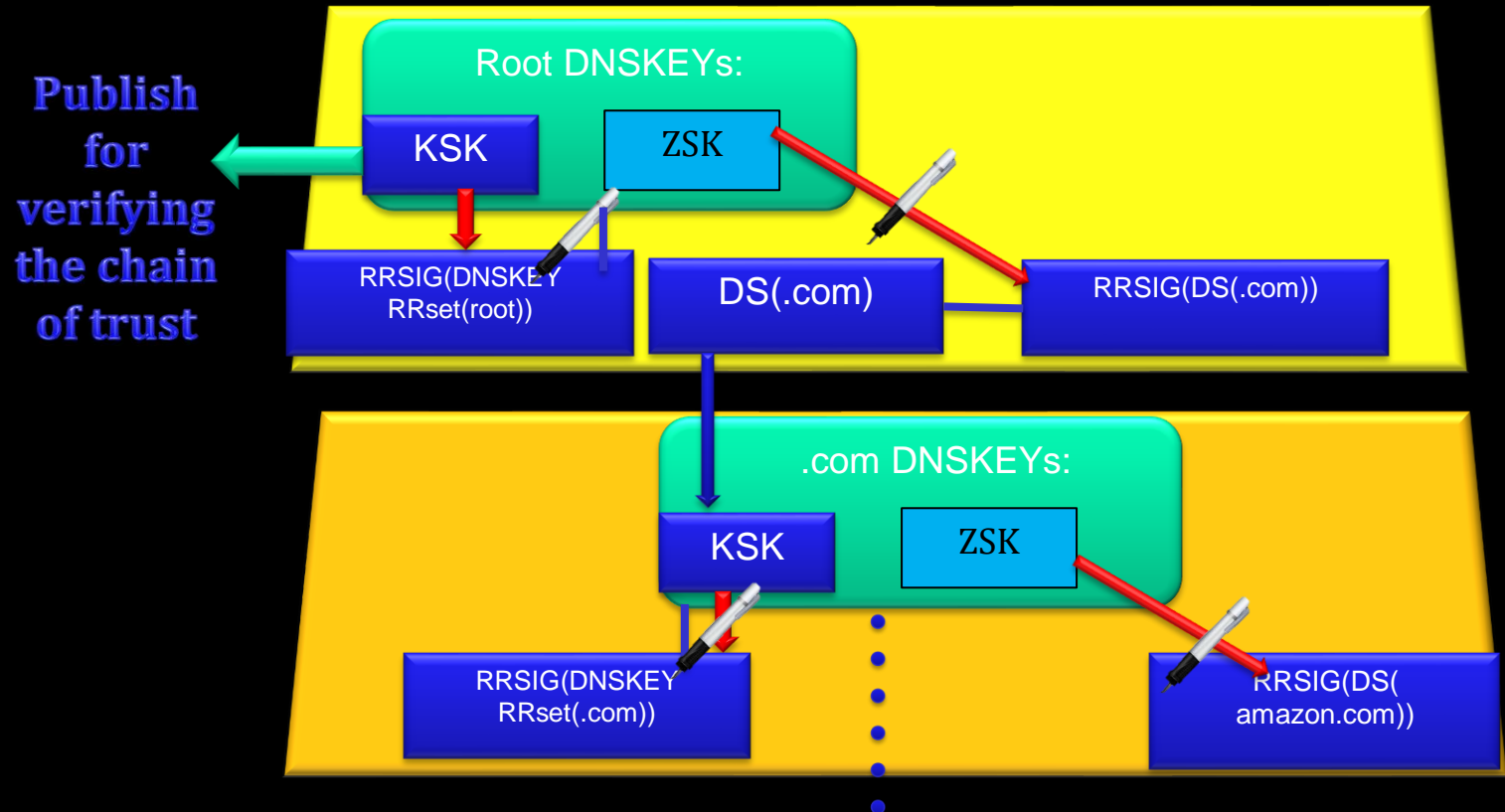
# PKI: chain of trust (2)

---

- **DNS query:**
  - **Public keys are stored in a new type of resource record, the DNSKEY RR**
    - **The private keys used to sign zone data must be kept secure**
  - **The target key has to be signed by either a configured authentication key or another key that has been authenticated previously**
  - **The target key: the public key is being used for authentication**
- **DS RR's used to link parent and child**
- **DS points to a Key Signing Key (KSK) of a child zone**
  - **Signature from that KSK over a DNSKEY RRset transfers trust to all keys in DNSKEY RRset**
  - **Key that DS points to, a KSK, only signs a DNSKEY RRset containing both KSK and ZSK**
- **Zone Signing Key (ZSK) in a DNSKEY RR sign entire zone's RR's**

# DNSKEY and DS

- KSK serves as the “anchor” of the authentication chain to a child zone
- Need to install at least one public key in a recursive server/resolver to anchor the authentication chain



# DS RR and RRSIG RR in parent zone (1)

---

- As part of the chain of trust, the zone has to inform its parent of its public key, KSK, securely through out-of-DNS channel means
  - The parent creates a hash of the public key of its child zone's KSK and stores it in the parent zone in a RR called a DS RR
  - It also signs this DS RR by generating a RRSIG RR
    - The keys periodically have to be changed because any key can be broken with sufficient computing power, aided by the volume of signature data generated
  - In a chained secure zone, whenever a zone changes its KSK, its parent has to be notified of the new key
  - The parent then has to generate a new DS RR and sign it again
- To reduce the administrative burden involved, a common strategy is to use another key pair, the ZSK for signing the child zone

# Separating the functions of KSK and ZSK

---

- Separating the functions of KSK and ZSK has several advantages:
  - No parent/child interaction is required when ZSKs are updated
  - The KSK can be made stronger (i.e., using more bits in the key material)
    - This has little operational impact since it is only used to sign a small fraction of the zone data
    - The KSK is only used to verify the zone's key set, not for other RRSets in the zone
    - As the KSK is only used to sign a key set, which is most probably updated less frequently than other data in the zone, it can be stored separately from, and in a safer location, than the ZSK
    - A KSK can have a longer key effective period

# Authentication Chain (1)

- A sequence of a ZSK in a DNSKEY RR and Delegation Signer (DS) RR in a parent zone, as well as the KSK in a child zone certified by the corresponding DS RR, forms an authentication chain of signed data
  - A DNSKEY RR (ZSK) is used to verify the signature covering a DS RR and allows the DS RR to be authenticated in a parent zone
    - The DS RR contains a hash of the KSK of a child zone and this KSK's DNSKEY RR is authenticated by matching the hash in the DS RR in the parent zone
      - This child zone KSK authenticates the DNSKEY RRset, which contains a ZSK, which in turn authenticates another DS RR, and so forth until the chain finally ends with a DNSKEY RR whose corresponding private key signs the desired DNS RR data
- Example
  - The root ZSK in a DNSKEY RR of the root zone is used to sign the DS RR for ".com"
  - The ".com" DS RRset contains a hash that matches ".com" KSK
  - This KSK signs the DNSKEY RRset, containing ZSK
  - The ZSK's private key signs the amazon.com's NS RRset
  - . . . .



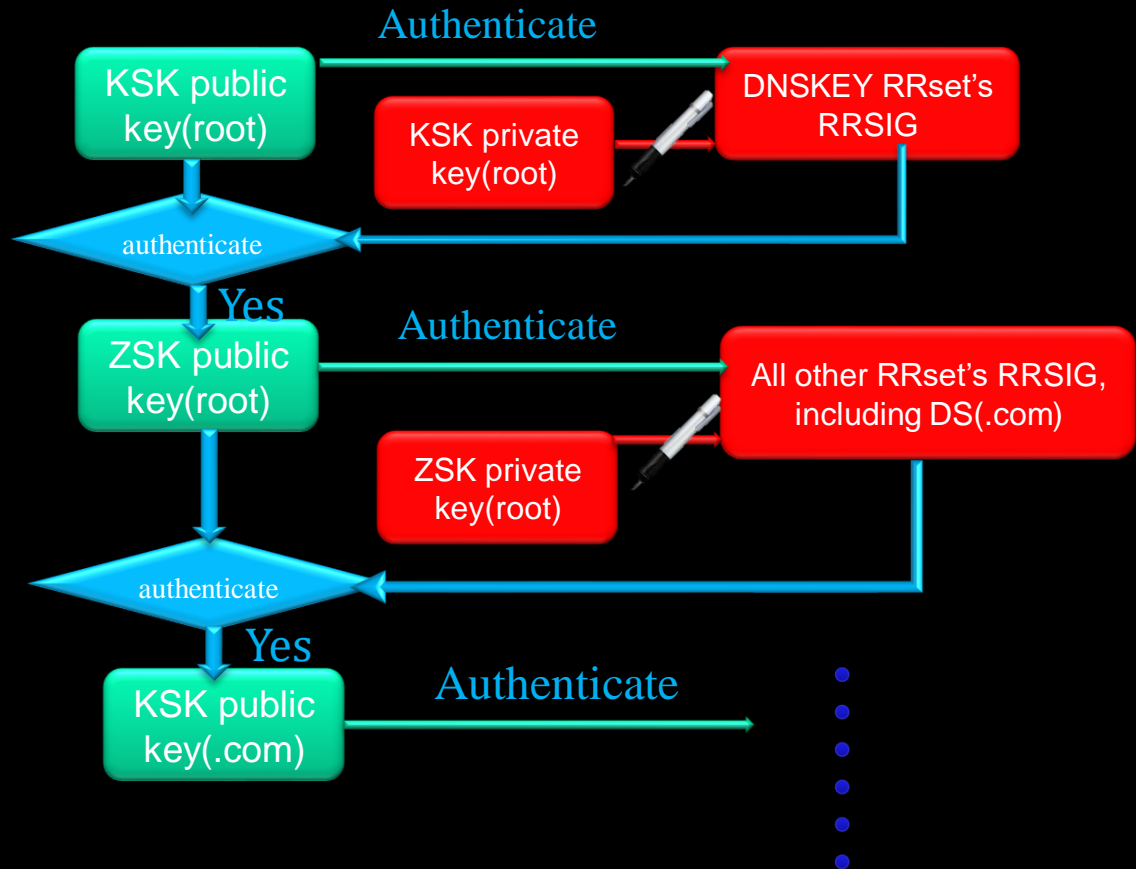
# Authentication Chain (2)

---

- **Example**
  - the root ZSK in a DNSKEY RR of the root zone is used to sign the DS RR for ".com"
  - The ".com" DS RR contains a hash that matches ".com" KSK
  - This KSK signs the DNSKEY RRset of ".com", containing ZSK
  - The ZSK's private key signs the amazon.com's NS RRset, DS(amazon.com) RR, ....
  - The amazon.com DS RR contains a hash that matches amazon.com's KSK
  - This amazon.com's KSK signs the DNSKEY RRset of "amazon.com", containing ZSK
  - The amazon.com's ZSK signs the amazon.com's RR's, including ww.amazon.com RR
- The root KSK is published for verifying the root ZSK....

# Authentication Chain using KSK(root)

- Signatures are pre-generated using private keys
- Authentication using public keys, starting from the anchor KSK(root)
- The public key of KSK(.com) is obtained using DNSKEY(.com) RR
  - The authentication for the public key of KSK(.com) uses the RRSIG(DS(.com))



# DNSSEC Deployment (1)

---

- **Feb. 28, 2009:**
  - The US government has digitally signed the .gov TLD, effectively implementing the Domain Name System Security Extensions (DNSSEC) protocols throughout the top tier of the federal Internet space
- **On 5/5/2010:**
  - The 13 authoritative root servers for the domain name system have switched to the DNS Security Extensions (DNSSEC) security protocol. All 13 root servers are now serving a signed version of the root zone.
- **DNSSEC in the .org TLD registry in June 2010**
- **DNSSEC in the .edu TLD registry in June 2010**
- **DNSSEC in the .net TLD registry in 12/31/2010**

## DNSSEC Deployment (2)

---

- The .com domain's DNSSEC became operational 3/31/2011
- The three largest zones are .com, .net, .org
  - The .com domain: the Internet's most popular top-level domain with more than 80 million registered names
  - The .org space has more than 7.5 million domains registered in it
  - The .gov top-level domain has about 3,700 domains
- Source: [https://www.dnssec-deployment.org/wp-content/uploads/2010/08/TLD-deployment-Table-8\\_30\\_10.pdf](https://www.dnssec-deployment.org/wp-content/uploads/2010/08/TLD-deployment-Table-8_30_10.pdf)

# Root Zone Signing (1)

---

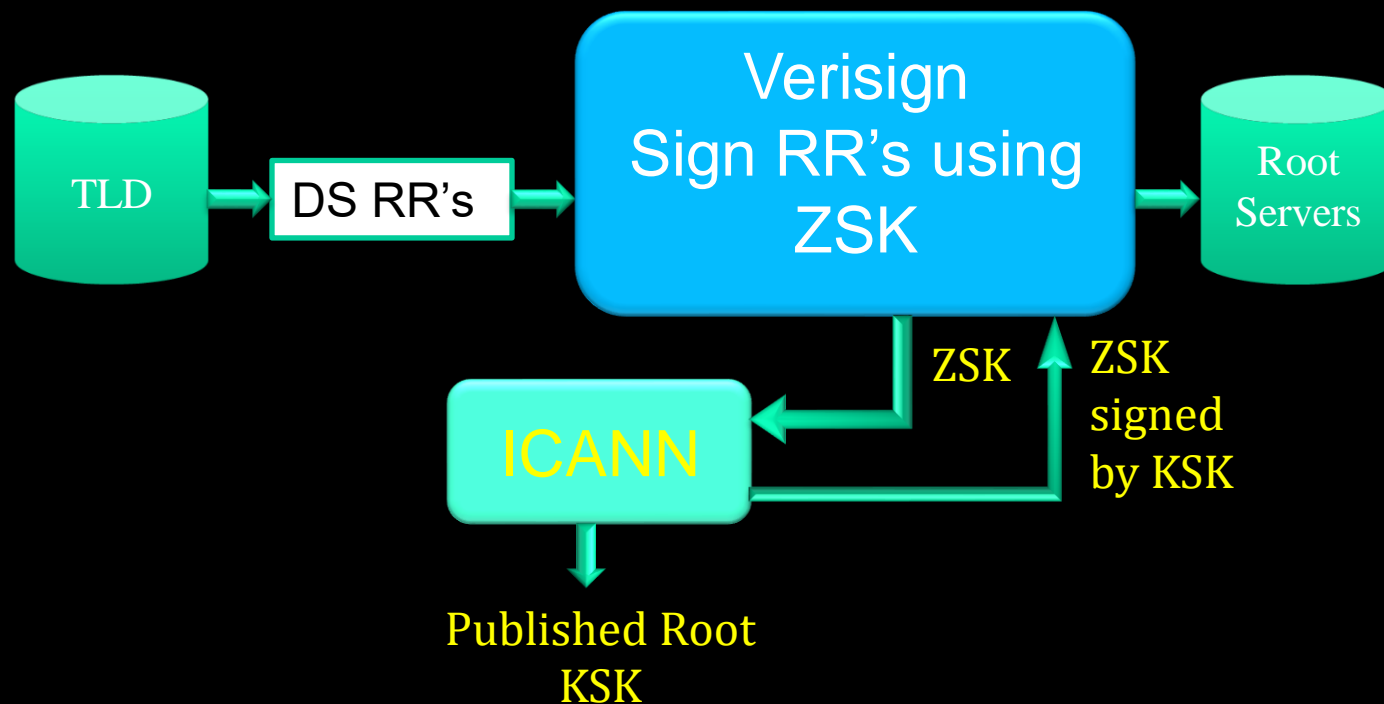
- **VeriSign is the Root Zone Maintainer**
  - **Manages the Root Zone Signing Key (ZSK)**
    - **1024 bits**
    - **ZSK is replaced four times a year (1-3 months)**
    - **US Government**
      - **RSA-SHA1 or RSASHA-256 until 2015**
      - **ECDSA after 2015**
  - **Signs the root zone with the ZSK**
  - **Distributes the signed zone to the root server operators**

# Root Zone Signing (2)

---

- Key Signing Key (KSK) is used to sign ZSK
  - 2048 bits
  - KSK is replaced one time a year (1-2 years)
  - US Government
    - RSA-SHA1 or RSASHA-256 until 2015
    - ECDSA after 2015
- ICANN publishes the public part of the KSK
- IANA Functions Operator
  - Manages the Key Signing Key (KSK)
  - Accepts DS records from TLD operators
  - Verifies and processes request
  - Sends update requests to DoC for authorization and to VeriSign for implementation

# Root Zone Signing



---

# **CSci530: Security Systems**

**Lecture 13 – November 18 2022**

## **Trusted Computing**

**Dr. Clifford Neuman**

**University of Southern California**

**Information Sciences Institute**



# **Trusted vs. Trustworthy**

---

- **We trust our computers**
  - **We depend upon them.**
  - **We are vulnerable to breaches of security.**
- **Our computer systems today are not worthy of trust.**
  - **We have buggy software**
  - **We configure the systems incorrectly**
  - **Our user interfaces are ambiguous regarding the parts of the system with which we communicate.**

# **A Controversial Issue**

---

- **Many individuals distrust trusted computing.**
- **One view can be found at <http://www.lafkon.net/tc/>**
  - **An animated short film by Benjamin Stephan and Lutz Vogel**

# **What is Trusted Computing**

---

- **Attestation**
  - Includes Trusted path
- **Separation**
  - Secure storage (data/keys)
  - Protection of processes
- **The rest is policy**
  - That's the hard part
  - And the controversial part

# **Separation of Security Domains**

---

- **Need to delineation between domains**
  - **Old Concept:**
    - **Rings in Multics**
    - **System/Privileged vs. User mode**
  - **But who decides what is trusted**
    - **User in some cases**
    - **Third parties in others**
    - **Trusted computing provides the basis for making the assessment.**

# Trusted Path

---

- We need a “trusted path”
  - For user to communicate with a domain that is trustworthy.
    - Usually initiated by escape sequence that application can not intercept: e.g. CTL-ALT-DEL
  - Could be direct interface to trusted device:
    - Display and keypad on smartcard

# Communicated Assurance

---

- We need a “trusted path” across the network.
- Provides authentication of the software components with which one communicates.

# **The Landscape – Early Work**

---

- **Multics System in late 1960s.**
  - **Trusted path, isolation.**
- **Paper on Digital Distributed System Security Architecture by Gasser, Goldstein, Kauffman, and Lampson.**
  - **Described early need for remote attestation and how accomplished.**

# **The Landscape – Industry**

---

- **Industry interest in the late 1990s.**
- **Consortia formed such as the Trusted Computing Group.**
- **Standards specifications, starting with specs for hardware with goal of eventual inclusion in all new computer systems.**
  - **Current results centered around attestation and secure storage.**



# **The Landscape – Applications**

---

- **Digital Rights Management**
- **Network Admission Control**
  - **PC Health Monitoring**
  - **Malware detection**
- **Virtualization of world view**
  - **VPN Segregation**
  - **Process control / SCADA systems**
- **Many other users**

## **Discussion - Risks**

---

- **Trusted computing is a tool that can be misused.**
  - **If one party has too much market power, it can dictate unreasonable terms and enforce them.**
- **Too much trust in trusted computing.**
  - **Attestation does not make a component trustworthy.**
  - **Some will rely too much on certifications.**

## **Discussion - Benefits**

---

- **Allows systems to be developed that require trustworthy remote components.**
  - **Provides protection of data when out of the hands of its owner.**
- **Can provides isolation and virtualization beyond local system.**
  - **Provides containment of compromise.**

## **Discussion – What's missing**

---

- **Tools to manage policy**
  - **Managing policy was limitation for TC support in Vista**
- **Applications that protect the end user**
  - **We need more than DRM and tools to limit what users run.**
- **New architectures and ways of thinking about security.**

# ~~CSci530: Security Systems~~ ■

**Lecture 14 – December 3<sup>rd</sup> 2021**

## **Trusted Computing (continued) then Privacy**

**Dr. Clifford Neuman**

**University of Southern California**

**Information Sciences Institute**

## **What can we do with TC?**

---

- **Clearer delineation of security domains**
  - **We can run untrusted programs safely.**
    - **Run in domain with no access to sensitive resources**
      - **Such as most of your filesystem**
      - **Requests to resources require mediation by TCB, with possible queries user through trusted path.**

## **Red / Green Networks (1)**

---

- **Butler Lampson of Microsoft and MIT suggests we need two computers (or two domains within our computers).**
  - **Red network provides for open interaction with anyone, and low confidence in who we talk with.**
  - **We are prepared to reload from scratch and lose our state in the red system.**

## **Red / Green Networks (2)**

---

- **The Green system is the one where we store our important information, and from which we communicate to our banks, and perform other sensitive functions.**
  - **The Green network provides high accountability, no anonymity, and we are safe because of the accountability.**
  - **But this green system requires professional administration.**
  - **My concern is that a breach anywhere destroys the accountability for all.**



# **Somewhere over the Rainbow**

---

- But what if we could define these systems on an application by application basis.
  - There must be a barrier to creating new virtual systems, so that users don't become accustomed to clicking "OK".
  - But once created, the TCB prevents the unauthorized retrieval of information from outside this virtual system, or the import of untrusted code into this system.
  - Question is who sets the rules for information flow, and do we allow overrides (to allow the creation of third party applications that do need access to the information so protected).

# Today's Systems Less Secure

- Functional requirements for today's distributed applications eliminate isolation.
  - Larger attack surface – applications and server interfaces reachable through the Internet.
  - Users demand instant access to their data from all devices, wherever they may be.
  - Users demand ability to move data between applications.
- But not all “applications” should allow this much sharing.
  - We need to restore isolation, but along functional boundaries.



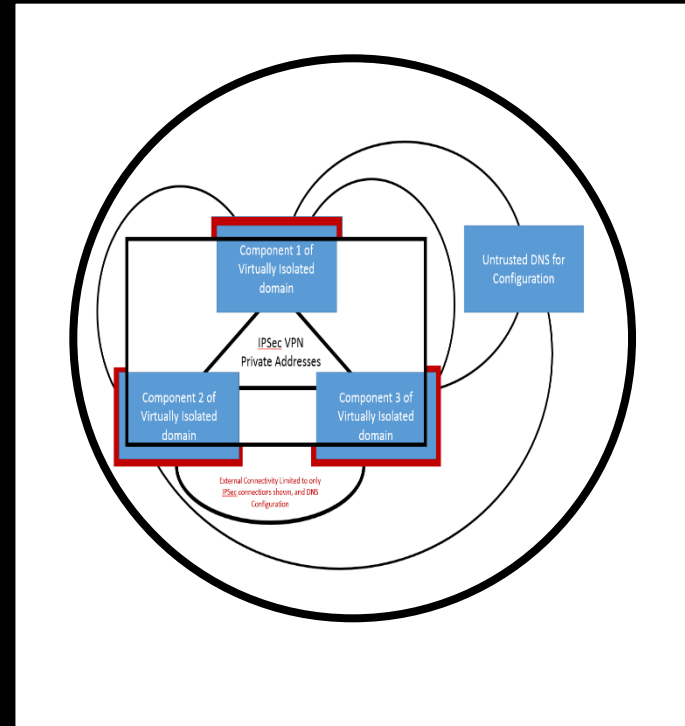
# **Many existing technologies *support* isolation**

- **Within computer systems**
  - Virtual Memory
  - Virtualization
  - Trusted computing
  - Data Encryption
- **Within Computer Networks**
  - Firewalls
  - Virtual Private Networks
  - Communication encryption
- **But our policies are too complex**
  - **Because they support isolation and sharing.**



# Changing our Concept of Isolation

- Changing the way we think of isolation
  - Not about artificial physical boundaries that are artifacts of how we build our systems
  - But rather around virtual boundaries that map onto the conceptual functions for which we use the systems.



# **What do we need for TC**

---

- **Trust must be grounded**
  - **Hardware support**
    - **How do we trust the hardware**
    - **Tamper resistance**
      - **Embedded encryption key for signing next level certificates.**
    - **Trusted HW generates signed checksum of the OS and provides new private key to the OS**

# Privacy of Trusted Hardware

---

- **Consider the processor serial number debate over Intel chips.**
  - **Many considered it a violation of privacy for software to have ability to uniquely identify the process on which it runs, since this data could be embedded in protocols to track user's movements and associations.**
  - **But Ethernet address is similar, although software allows one to use a different MAC address.**
  - **Ethernet addresses are often used in deriving unique identifiers.**

# **The Key to your Trusted Hardware**

---

- Does not have to be unique per machine, but uniqueness allows revocation if hardware is known to be compromised.
  - But what if a whole class of hardware is compromised, if the machine no longer useful for a whole class of applications. Who pays to replace it.
- A unique key identifies specific machine in use.
  - Can a signature use a series of unique keys that are not linkable, yet which can be revoked (research problem).

# **Non-Maskable Interrupts**

---

- **We must have hardware support for a non-maskable interrupt that will transfer program execution to the Trusted Computing Base (TCB).**
  - **This invokes the trusted path**



# **The Hardware Basis**

---

- **Trusted computing is proof by induction**
  - **Each attestation stage says something about the next level**
  - **Just like PKI Certification hierarchy**
- **One needs a basis step**
  - **On which one relies**
  - **Hardware is that step**
    - **(well, second step anyway)**

# Hardware Topics

---

- Trusted Platform Module
- Discussion of Secure Storage
- Boot process

# Trusted Platform Module

---

- **Basically a Key Storage and Generation Device**
- **Capabilities:**
  - **Generation of new keys**
  - **Storage and management of keys**
    - **Uses keys without releasing**

# Trusted Platform Module (TPM)?

---

## Smartcard-like module on the motherboard that:

- Performs cryptographic functions
  - RSA, SHA-1, RNG
  - Meets encryption export requirements
- Can create, store and manage keys
  - Provides a unique Endorsement Key (EK)
  - Provides a unique Storage Root Key (SRK)
- Performs digital signature operations
- Holds Platform Measurements (hashes)
- Anchors chain of trust for keys and credentials
- Protects itself against attacks



TPM 1.2 spec:  
[www.trustedcomputinggroup.org](http://www.trustedcomputinggroup.org)

**Slide From Steve  
Lamb at Microsoft**

# Why Use A TPM?

---

- **Trusted Platforms use Roots-of-Trust**
  - A TPM is an implementation of a Root-of-Trust
- **A hardware Root-of-Trust has distinct advantages**
  - **Software can be hacked by Software**
    - Difficult to root trust in software that has to validate itself
  - **Hardware can be made to be robust against attacks**
    - Certified to be tamper resistant
  - **Hardware and software combined can protect root secrets better than software alone**
- **A TPM can ensure that keys and secrets are only available for use when the environment is appropriate**
  - **Security can be tied to specific hardware and software configurations**

**Slide From Steve  
Lamb at Microsoft**

# Endorsement Key

---

- Every TPM has unique Endorsement key
  - Semi-root of trust for system
  - Generated and installed during manufacture
    - Issues
  - Real root is CA that signs public key associated with Endorsement key

# Using Encryption for Atestation

---

- **Extend**
  - Add data to a PCR
  - 20 byte hash hashed into current PCR
  - As each module loaded its hash extends the PCR
- **Quote**
  - Sign current value of PCR

# Secure Storage

---

- **Full Disk Encryption**
  - Key in register in disk
  - Or key in TPM and data encrypted/decrypted by TPM
- **Seagate Drive uses register in Disk**
  - Key must be loaded
  - User prompt at BIOS
  - Or managed by TPM
    - But OS image maybe on disk, how to get



# **OS Support for Trusted Computing (1)**

---

- **Separation of address space**
  - **So running processes don't interfere with one another.**
- **Key and certificate management for processes**
  - **Process tables contain keys or key identifiers needed by application, and keys must be protected against access by others.**
  - **Processes need ability to use the keys.**

## **OS Support for Trusted Computing (2)**

---

- **Fine grained access controls on persistent resources.**
  - **Protects such resources from untrusted applications.**
- **The system must protect against actions by the owner of the system.**

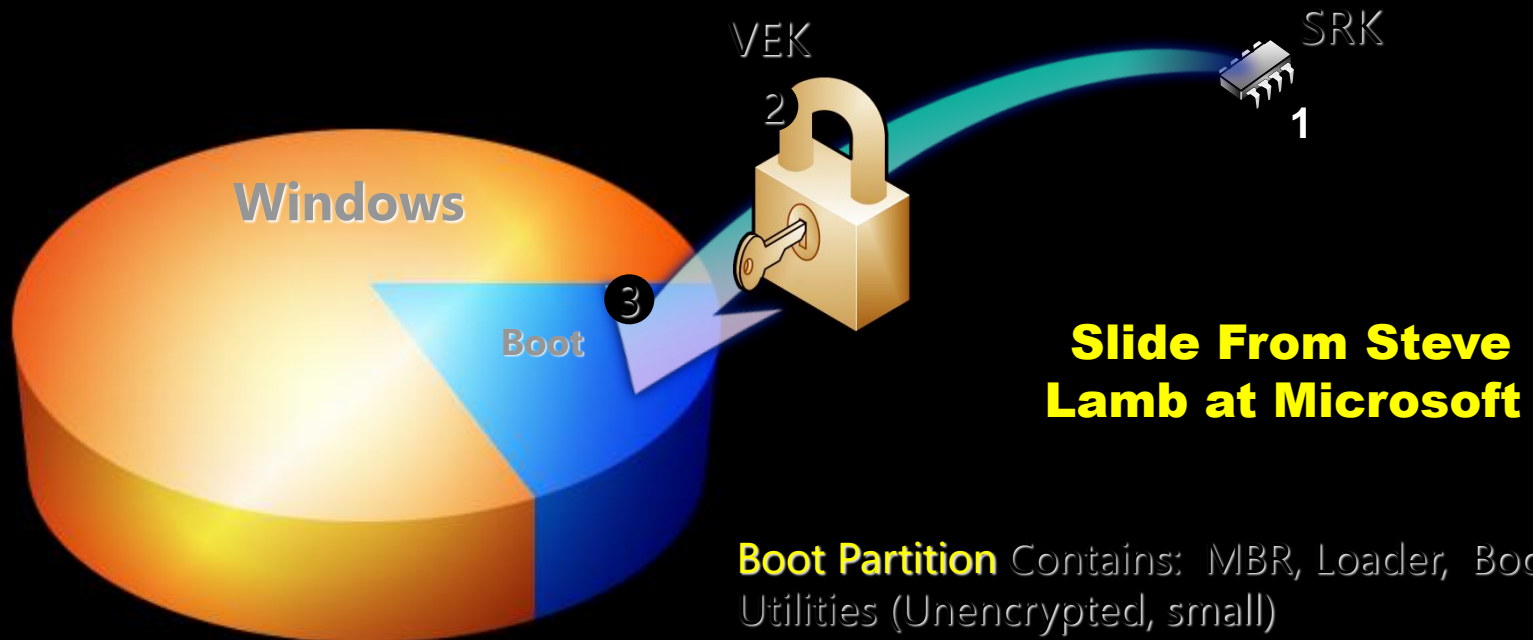
# Disk Layout & Key Storage

## Windows Partition Contains

- Encrypted OS
- Encrypted Page File
- Encrypted Temp Files
- Encrypted Data
- Encrypted Hibernation File

## Where's the Encryption Key?

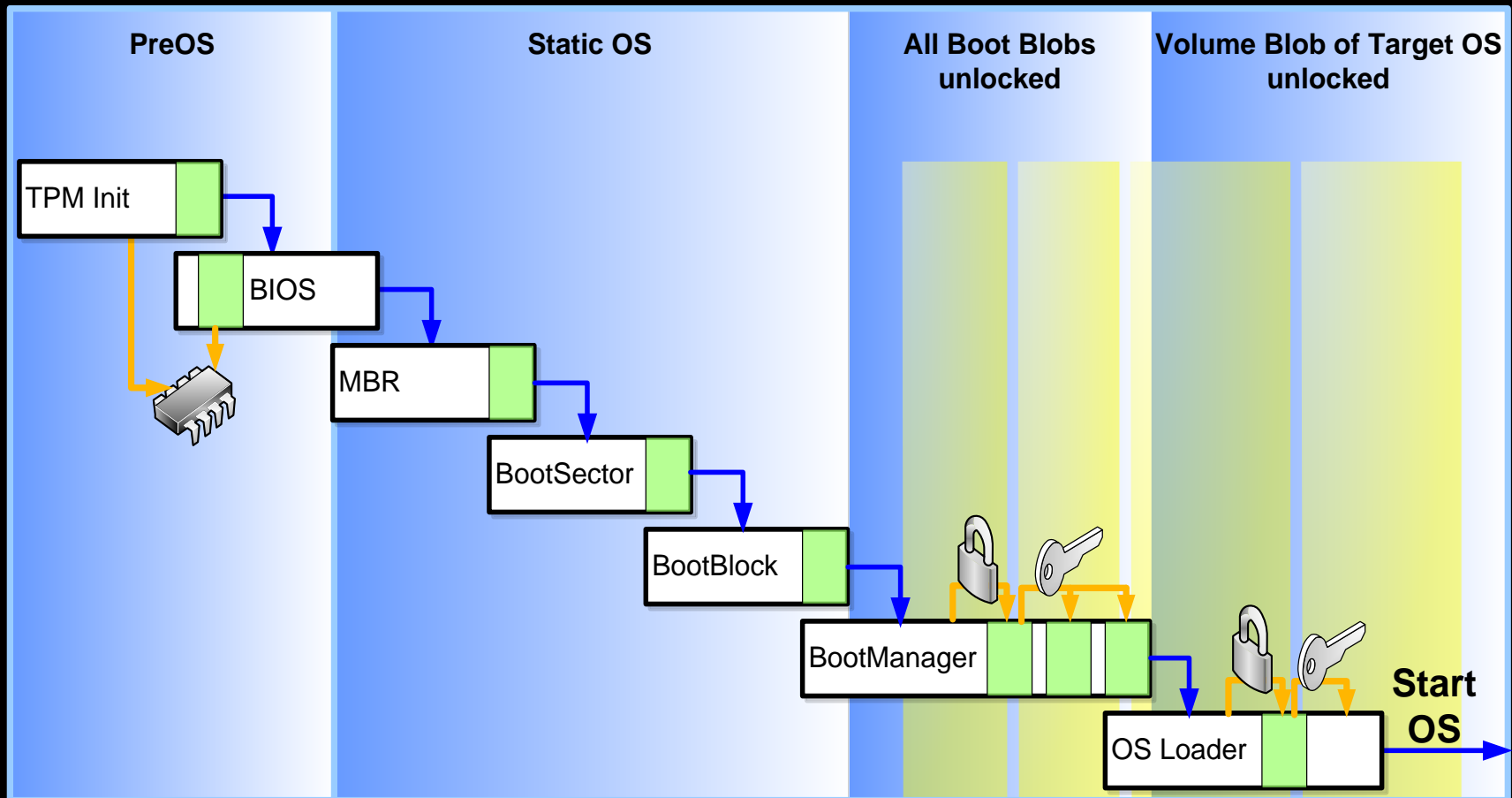
1. **SRK** (Storage Root Key) contained in **TPM**
2. **SRK** encrypts **VEK** (Volume Encryption Key) protected by **TPM/PIN/Dongle**
3. **VEK** stored (encrypted by **SRK**) on hard drive in Boot Partition



# BitLocker™ Architecture

Static Root of Trust Measurement of early boot components

Slide From Steve Lamb at Microsoft



---

# **CSci530: Security Systems**

## **Lecture 14 – December 2, 2022**

### **Privacy**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

# Outline of Discussion

---

- Introduction – security vs privacy
- You are being tracked
- Aggregation of data
- Traffic analysis and onion routing
- P3P and Privacy Statements
- Protecting data on personal laptops/desktops
- Forensics
- Retention/Destruction Policies
- Who's data is it anyway

# What is Privacy?

---

- Privacy is about Personally Identifiable Information
- It is primarily a policy issue
  - Policy as a system issue
    - Specifying what the system should allow
  - Policy as in public policy
    - Same idea but less precise and must be mapped
- Privacy is an issue of user education
  - Make sure users are aware of the potential use of the information they provide
  - Give the user control
- Privacy is a Security Issue
  - Security is needed to implement the policy

# Security v. Privacy

---

- Sometimes conflicting
  - Many security technologies depend on identification.
  - Many approaches to privacy depend on hiding ones identity.
- Sometime supportive
  - Privacy depends on protecting PII (personally identifiable information).
  - Poor security makes it more difficult to protect such information.



# **Major Debate on Attribution**

---

- **How much low level information should be kept to help track down cyber attacks.**
  - **Such information can be used to breach privacy assurances.**
  - **How long can such data be kept.**

# **Privacy not Only About Privacy**

---

- **Business Concerns**
  - **Disclosing Information we think of as privacy related can divulge business plans.**
    - **Mergers**
    - **Product plans**
    - **Investigations**
- **Some “private” information is used for authentication.**
  - **SSN**
  - **Credit card numbers**

# **You Are Being Tracked**

---

- **Location**
  - From IP address
  - From Cell Phones
- **Interests, Purchase History, Political/Religious Affiliations**
  - From Transaction Details
  - From network and server traces
- **Associates**
  - From network, phone, email records
  - From location based information
- **Health Information**
  - From Purchases
  - From Location based information
  - From web history

## 2009 current event

- New York Times – Miguel Helft – November 11 2008.

---

- SAN FRANCISCO — There is a new common symptom of the flu, in addition to the usual aches, coughs, fevers and sore throats. Turns out a lot of ailing Americans enter phrases like “flu symptoms” into Google and other search engines before they call their doctors.  
– link

# Why Should you Care?

---

- Aren't the only ones that need to be concerned about privacy the ones that are doing things that they shouldn't?
- Consider the following:
  - Use of information outside original context
    - Certain information may be omitted
  - Implications may be mis-represented.
  - Inference of data that is sensitive.
  - Such data is often not protected.
  - Data can be used for manipulation.

# Old News - Shopper's Suit Thrown Out

## Los Angeles Times – 2/11/1999

---

- Shopper's Suit Thrown Out
- By [Stuart Silverstein](#), Staff Reporter  
[February 11, 1999](#) *in print edition C-2*
- A Vons shopper's lawsuit that raised questions about the privacy of information that supermarkets collect on their customers' purchases has been thrown out of court. Los Angeles Superior Court Judge David Horowitz tossed out the civil suit by plaintiff Robert Rivera of Los Angeles, declaring that the evidence never established that Vons was liable for damages.
- The central issue in the case was a negligence claim Rivera made against Vons. It stemmed from an accident at the [Lincoln Heights](#)' Vons in 1996 in which Rivera slipped on spilled yogurt and smashed his kneecap.
- Although that issue was a routine legal matter, the case drew attention because Rivera raised the privacy issue in the pretrial phase. Rivera claimed that he learned that Vons looked up computer records of alcohol purchases he made while using his club discount card and threatened to use the information against him at trial.
- Vons, however, denied looking up Rivera's purchase records and the issue never came up in the trial, which lasted two weeks before being thrown out by the judge Tuesday.
- A Vons spokesman said the company was "gratified by the judge's decision." M. Edward Franklin, a Century City lawyer representing Rivera, said he would seek a new trial for his client.

# Aggregation of Data

---

- Consider whether it is safe to release information in aggregate.
  - Such information is presumably no longer personally identifiable
  - But given partial information, it is sometimes possible to derive other information by combining it with the aggregated data.

# Anonymization of Data

---

- Consider whether it is safe to release information that has been stripped of so called personal identifiers.
  - Such information is presumably no longer personally identifiable
  - But is it. Consider the release of AOL search data that had been stripped of information identifying the individual performing the search.
    - What is important is not just anonymity, but likability.
    - If I can link multiple queries, I might be able to infer the identity of the person issuing the query through one query, at which point, all anonymity is lost.



# Traffic Analysis

---

- Even when specifics of communication are hidden, the mere knowledge of communication between parties provides useful information to an adversary.
  - E.g. pending mergers or acquisitions
  - Relationships between entities
  - Created visibility of the structure of an organizations.
  - Allows some inference about your interests.

# **Obama's cell phone records breached**

## **Washington (CNN) 11/21/2008**

---

- Records from a cell phone used by President-elect Obama were improperly breached, apparently by employees of the cell phone company, Verizon Wireless said Thursday.
- "This week we learned that a number of Verizon Wireless employees have, without authorization, accessed and viewed President-Elect Barack Obama's personal cell phone account," Lowell McAdam, Verizon Wireless president and CEO, said in a statement.
- McAdam said the device on the account was a simple voice flip-phone, not a BlackBerry or other smartphone designed for e-mail or other data services, so none of Obama's e-mail could have been accessed.
- Gibbs said that anyone viewing the records likely would have been able to see phone numbers and the frequency of calls Obama made, but that "nobody was monitoring voicemail or anything like that."

# Linkages – The Trail We Leave

---

- **Identifiers**
  - **IP Address**
  - **Cookies**
  - **Login IDs**
  - **MAC Address and other unique IDs**
  - **Document meta-data**
  - **Printer microdots**
- **Where saved**
  - **Log files**
- **Persistence**
  - **How often does Ip address change**
  - **How can it be mapped to user identification**

# Unlinking the Trail

---

- **Blind Signatures**
  - Enable proof of some attribute without identifying the prover.
  - Application in anonymous currency.
  - Useful in voting.

# Unlinking the Trail

---

- **Anonymizers**
  - A remote web proxy.
  - Hides originators IP address from sites that are visited.
  - Usually strips off cookies and other identifying information.
- **Limitations**
  - You are dependent on the privacy protections of the anonymizer itself.
  - All your activities are now visible at this single point of compromise.
  - Use of the anonymizer may highlight exactly those activities that you want to go unnoticed.

# Onion Routing

---

- **Layers of peer-to-peer anonymization.**
  - You contact some node in the onion routing network
  - Your traffic is forward to other nodes in the network
  - Random delays and reordering is applied.
  - With fixed probability, it is forwarded on to its destination.
- **TA requires linking packets through the full chain of participants.**
  - And may be different for each association.

# Protecting Data in Place

---

- Many compromises of privacy are due to security compromised on the machines holding private data.
  - Your personal computer or PDAs
  - Due to malware or physical device theft
- Countermeasures
  - For device theft, encryption is helpful
  - For malware, all the techniques for defending against malicious code are important.
  - Live malware has the same access to data as you do when running processes, so encryption might not be sufficient.

# Forensics

---

- Tools are available to recover supposedly deleted data from disks.
  - Similar tools can reconstruct network sessions.
  - Old computers must be disposed of properly to protect any data that was previously stored.
    - Many levels of destruction
  - Tools like whole disk encryption are useful if applied properly and if the keys are suitably destroyed.



# **Privacy – Retention Policies**

---

- **PII (personally identifiable information)**
  - **Is like toxic waste**
  - **Don't keep it if you can avoid it**
- **Regulations**
  - **Vary by Jurisdiction**
  - **But if you keep it, it is “discoverable”**



# Securing the Smart Grid

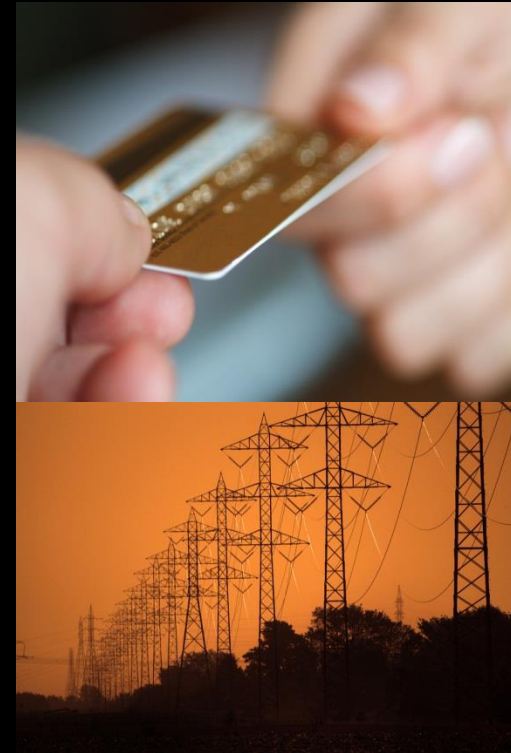
- Prof. Clifford Neuman  
Director, USC Center  
for Computer System  
Security

UNIVERSITY OF SOUTHERN CALIFORNIA  
**INFORMATION  
SCIENCES  
INSTITUTE**

# Critical Infrastructure

---

- **Critical**
  - Compromise can be catastrophic
  - Existing approaches to protection often based on isolation.
- **Infrastructure**
  - It touches everything
  - It can't be isolated by definition
- **Smart (or i- or e-)**
  - We cant understand it
  - The Cyber Components cant be isolated.



# Outline

---

- **The Power Grid is Federated**
  - Even more so for the Smart Grid
- **Security Depends on Defining Boundaries**
  - Cyber and Physical Domains
- **Resiliency of the Power Grid**
  - Operational Resiliency
  - Resiliency of Individual Functions
- **Using Smarts to Improve Resiliency**
  - Redefining Boundaries



# **Trends in Power Systems**

---

- **Evolution of power distribution**
  - **Local power systems**
  - **Interconnected**
  - **More centralized control**
  - **Automated reaction to events**
  - **Reaching into the neighborhoods**
  - **Encompassing the home**

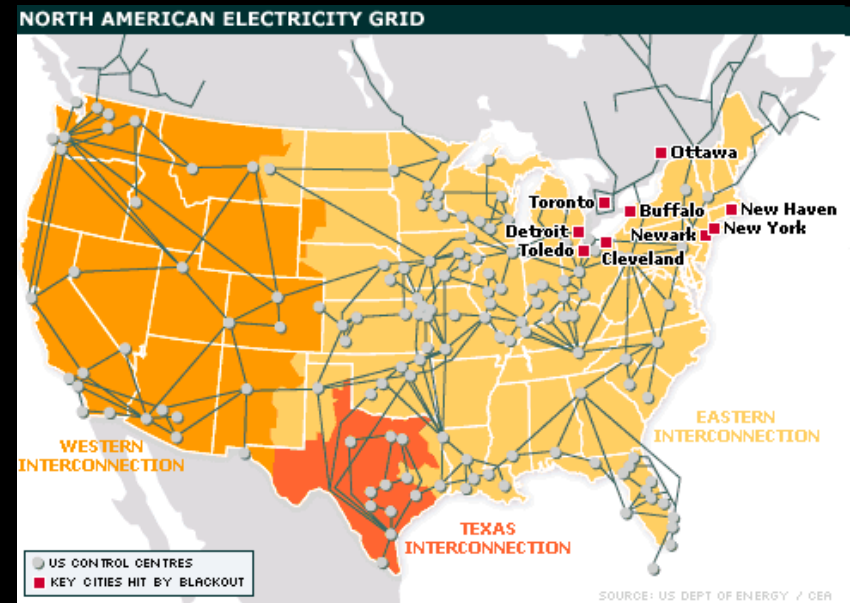
# Federated Systems

---

- **Characteristics**
  - Parts of the system managed by different parties.
  - Lack of physical control in placement of components
  - Lack of a common set of security policies
  - The “administrative” dimension of scalability
- **The Power Grid is Naturally Federated**
  - **Other Federated Systems**
    - The Financial System
    - Cloud Computing
    - The Internet in general

# Federation in Power Systems

- Power systems span large geographic areas and multiple organizations.
  - Such systems are naturally federated
- Avoiding cascading blackouts requires increasingly faster response in distant regions.
  - Such response is dependent on network communication.
- Regulatory, oversight, and “operator” organizations exert control over what once were local management issues.
  - Staged power alerts and rolling blackouts
- Even more players as the network extends to the home.
  - Customers
  - Information Providers



# **Meaning of Security for C-P Systems**

---

- In traditional cyber systems, emphasis on:
  - Confidentiality and Integrity
- In Cyber-Physical systems much greater emphasis on:
  - Resiliency (one example of “availability”)
  - The consequences of failure are greater.
- **Interrelation of Integrity with Resilience**



# Understanding Securability

---

- **Security is About Boundaries**
  - We must understand the boundaries
  - Containment of compromise is based on those boundaries
- **Federated Systems Cross Boundaries**
  - Federation is about control
    - And the lack of central coordinated control
    - By definition, we can't control parts of the system.
  - Protecting such systems requires constraints at the boundaries.

# Securing the Power Grid

---

- **Traditional Security**
  - It's about protecting the perimeter.
  - Imposing policy on ability to access protected resources.
- **In Federated Systems**
  - The adversary is within the perimeter.
  - There are conflicting policies.
- **The failure lies in not defining the perimeter**
  - Or more precisely, in choosing the wrong one
  - Allowing the boundaries to change
  - Not implementing correct containment at the boundary

# Threat Propagation

---

- Modeling can help us understand how threats propagate across domains.
  - **There are several classes of propagation to be considered, based on the domains that are crossed.**
    - **Cyber-Cyber**
    - **Cyber-Physical**
    - **Physical-Cyber**
    - **Physical-Physical**
    - **And transitive combinations.**

# Cyber-Cyber Threats

---

- Cyber-Cyber threats (traditional cyber security)
  - Easily scaled (scripts and programs)
  - Propagate freely in undefended domains
  - We understand basic defenses (best practices)

# Cyber-Physical Threats

---

- Cyber-Physical threats  
(physical impact of cyber activity)
  - **Implemented through PLC**
    - **or by PHC (social engineering)**
    - **or less direct means (computing power consumption)**
  - **Physical impact from programmed action**
  - **But which domain is affected (containment)**

# Physical-Cyber Threats

---

- Physical-Cyber threats (impact to computing)
  - For example, causing loss of power to or destruction of computing or communication equipment.
    - A physical action impacts the computation or communication activities in a system.
  - Containment through redundancy or reconfiguration
    - Standard disaster recovery techniques including off-site backup, and even cloud computing.
  - Still need to expect
    - Computing supply chain issues and hardware provenance (counterfeit products, or changes during fabrication).

# Physical-Physical Threats

---

- Physical-Physical threats (propagation of impact)
  - Traditionally how major blackouts occur
    - Cascading failure across domains
    - System follows physics, and effects propagate.
  - Containment is often unidirectional
    - A breaker keeps threat from propagating upward
    - But it explicitly imposes the impact downward
    - Firewalls and circuit breakers have analogies in many problem domains (including the financial sector)
    - Reserves often necessary for containment
  - Such containment in problem specific areas often protects against only known threats.

# Transitive Threats (example)

---

- Dependence on unsecure web sites as control channels.
  - End customer smart devices (including hybrid vehicles) will make decisions based on power pricing data.
    - Or worse – based on an iPhone app
  - What if the this hidden control channel is not secure
    - Such as a third party web site or
    - Smart Phone viruses
  - An attack such control channels could, for example, set pricing data arbitrarily high or low, increase or decrease demand, or directly controlling end devices.
    - Effectively cycling large number of end devices almost simultaneously.





# Transitive Threats

---

- More interesting real-world threats combine the binary threats for greater impact.
  - **Cyber-Physical-Physical**
    - Multiple Chevy Volts's controlled from hacked smartphones.
  - **Cyber-Physical-Cyber (CPC)**
    - Controlling device on HAN that causes meter to generate alerts creating DOS on AMI network.
  - **Physical-Cyber-Physical (PCP)**
    - Leverage Cyber response, e.g. 3 Sensor Threshold for fire suppression system.



# The Correct Perimeters

---

- **Systems can be secure for a particular function**
  - We need to define perimeters for particular functions
- **In the Power Grid**
  - Billing and Business operations are one function
  - SCADA and infrastructure control are another.
  - In the smart grid, customer access and HAN control a third

# Changing Boundaries

---

- Federated systems change over time
  - They evolve with new kinds of participants
    - E.g. Power grid → Smart Grid
    - Now the customer is part of the control loop
  - New peers join the federation
    - Not all may be as trusted
    - An adversary could acquire an existing participant
  - Mis-guided public policy could require expansion of protection domains.
  - This is why a monolithic security domain will not work.

# Containment

---

- **Containment techniques must be appropriate to the boundary and the function to be protected.**
  - **Firewalls, Application Proxies, Tunnels (VPN's) suitable in the Cyber Domain.**
  - **Cyber-Physical boundaries require different techniques.**
    - **We must understand cyber and physical paths**
    - **We must understand the coupled systems of systems impact of faults originating in single domain.**
    - **We must understand the C-P impact of Cyber attack automation**
  - **We need to group similar, yet distinct protection domains.**

# Understanding Resilience

---

- *Operational Resilience* is the capability of a system to fulfill its mission in a timely manner, even in the presence of attacks or failures.
  - The definition also usually includes the ability of the system to restore such capability, once it has been interrupted.
  - A system performs many functions and operational resilience is a function of functional resilience of different aspects of the system.
  - The function depends on domain understanding (especially time-scales)

# Smart Grids are More Resilient

---

- Automation and redundancy can mitigate impacts of failures within the system.
  - Multiple communications paths through Internet, and AMI
  - Demand response can provide new “reserve” capacity.
  - “Distributed Generation” may be closer to loads
  - Improvements in energy storage can increase timescales over which load and supply must be balanced.
- But reliance on these technologies makes the system more dependent on the communication and IT infrastructure
  - Which becomes a point of attack on the system.

## **How resilience used to be achieved**

---

- ***Availability*** has always been a critical service for power control networks and C-P systems
  - The control network for interconnects was managed separately.
    - Sole purpose was to exchange commands and information needed to keep the system functional.
  - Integrity and confidentiality was provided through limited physical access.

# Securing the Smart Grid

---

- We must recognize that complete physical separation is no longer possible
  - **Because the Smart Grid extends into physically unsecure areas.**
- Thus we must provide isolation through technical means.
  - We must define protection domains
  - **Improve support in the hardware, OS, and middleware to achieve isolation.**
  - **Design the system to identify policy on control flows so that Smart Grid components enforce it.**



# Summary

---

- The Smart Grid extends to homes & businesses
  - **New security implications for such connections.**
  - **Hidden control channels.**
- Critical and non-critical functions will not be separate
  - **Availability is critical – Defined as Resilience**
  - **Performance isolation needed for critical communication.**
- The federated nature of the smart grid demands:
  - **Federated architectures to secure it.**
  - **Federated systems to model it**
- Existing security for the power grid does not address the implications of the new architecture.
  - Containment Architecture is Needed
  - Many domains based on participants, and physical structure of the system.
- **Resiliency is Key**
  - Reconfigurability and Islanding can help



---

# **CSci530: Security Systems**

## **Lecture 14 – Additional Topics**

### **Security in the Cloud**

**Dr. Clifford Neuman**  
**University of Southern California**  
**Information Sciences Institute**

# Defining The Cloud

---

- The cloud is many things to many people
  - Software as a service and hosted applications
  - Processing as a utility
  - Storage as a utility
  - Remotely hosted servers
  - Anything beyond the network card
- Clouds are hosted in different ways
  - Private Clouds
  - Public Clouds
  - Hosted Private Clouds
  - Hybrid Clouds
  - Clouds for federated enterprises

# **Risks of Cloud Computing**

- **Reliability**
  - Must ensure provider's ability to meet demand and to run reliably
- **Confidentiality and Integrity**
  - Service provider must have their own mechanisms in place to protect data.
  - The physical machines are not under your control.
- **Back channel into own systems**
  - Hybrid clouds provide a channel into ones own enterprise
- **Less control over software stack**
  - Software on cloud may not be under your enterprise control
- **Harder to enforce policy**
  - Once data leaves your hands

# Cloud Security Summary

---

- **Great potential for cloud computing**
  - Economies of scale for managing servers
  - Computation and storage can be distributed along lines of a virtual enterprise.
  - Ability to pay for normal capacity, with short term capacity purchases to handle peak needs.
- **What needs to be addressed**
  - Forces better assessment of security requirements for process and data.
  - Accreditation of providers and systems is a must.
  - Our models of the above must support automated resolution of the two.

---

# REVIEW

# Final Exam

---

- Open Book
- Open Note
- Electronics Allowed
- 120 Minutes (11AM to 1PM)
- Monday December 13<sup>th</sup>
- Online Only (no classroom available)

# **Review - Topics**

---

- **Cryptography**
- **Key Management**
- **Identity Management (and Authentication)**
- **Policy (and Authorization)**
- **Attacks**
  - **Classic**
  - **The human element**
- **Defenses**
  - **Firewalls, Intrusion Detection and Response, Encryption, Tunnels, Defenses to Malware**
- **Architectures and Trusted Computing**
- **Cyber-Physical and Cloud Computing**



# **Glossary of Attacks**

---

**This is not a complete list**

- **Availability**
  - **Denial of Service (DoS AND DDoS)**
    - **Over consumption of resources**
      - **Network, ports, etc**
      - **Take down name servers, other critical components**
    - **Exploits to crash system**
    - **Cache poisoning**

# **Glossary of Attacks**

---

**This is not a complete list**

- **Confidentiality**
  - **Eavesdropping**
  - **Key Cracking**
  - **Exploiting Key Mismanagement**
  - **Impersonation**
    - **Exploiting protocol weakness**
    - **Discovered passwords**
    - **Social Engineering**
  - **Exploiting mis-configurations**

# **Glossary of Attacks**

---

**This is not a complete list**

- **Integrity**
  - **Breaking Hash Algorithms**
  - **Exploiting Key Mismanagement**
  - **Impersonation**
    - **Exploiting protocol weakness**
    - **Discovered passwords**
    - **Social Engineering**
  - **Exploiting mis-configurations**
  - **Cache Poisoning**

# **Glossary of Attacks**

---

**This is not a complete list**

- **Miscellaneous**
  - **Spam**
  - **Phishing**
  - **Malware attacks**
    - **Spyware**
    - **Viruses**
    - **Worms**
    - **Trojan Horse**
  - **Man in the middle**
  - **SQL Injection**
  - **Cross Site Scripting**

# 2013 Final Exam – Q1

---

## Intrusion Detection and Trusted Computing

- a) Why is signature-based intrusion detection poorly suited for the detection of zero-day attacks? (10 points)
- b) What are the strengths of a network-based collector in an intrusion detection system? (5 points)
- c) What are the strengths of a host- or application-based collector in an intrusion detection system? (5 points)
- d) What is the difference between attestation and accreditation? (10 points)
- e) Explain what it means to “Extend a PCR”? (5 points)
- f) What is the function of the “endorsement key”, and how do we know that the correct endorsement key was used for the claimed function? (5 points)

# 2013 Final Exam – Q2

---

## Privacy and user Tracking

For each of the following techniques used to protect privacy or to breach user's privacy, match them with relevant terms or approaches used to either implement or defend against the technique. This is not a one-to-one mapping; more than one term may be relevant to a technique, and more than one technique may use the same term in its implementation or description. If you list a match that we are not looking for, but which is still correct, while you will not lose credit, you will not get credit either. You will lose a point if you associated an approach or technique with a threat that it is not effective against. There are more blanks in the page below than actual correct answers, so you do not need to fill in all the blanks.

- |   |  |
|---|--|
| 1. Traffic Analysis   | i. Cookie                              |
| 2. User tracking  | ii. Anonymization                      |
| 3. Data mining / inference  | iii. Onion Routing                     |
| 4. Spyware (including unexpected functions in installed software) | iv. User Education                     |
| 5. Linkability  | v. Personally Identifiable Information |
| 6. P3P, DoNotTrack, and Privacy Policies                          | vi. Encryption                         |
|   | vii. Aggregation                       |
|   | viii. User location                    |

# 2013 Final Exam – Q3.1

---

## Securing your own IT Infrastructure (40 points)

- You have a paranoid streak and have gotten tired of relying on service providers to secure your information. You are no longer willing to depend on someone else the cloud for backup and storage and you are determined to set up your own IT infrastructure to manage your own data. Fortunately, there are now a large number of products available that can assist you in doing just that. Unfortunately, many of these products leave some inherent vulnerability in your resulting system. In this problem, you are going to explore those issues and begin to understand just how hard it is to make your system truly secure.

# 2013 Final Exam – Q3.2

---

## The requirements for your system are:

- i. You will support a file system (or file systems) capable for storing at least 2 TB of data. Some of this data you consider to be highly sensitive (e.g. tax returns, credit card statements), some is critically sensitive such as passwords and encryption keys, while other data is less sensitive, and you will want to ability to share such less sensitive information with other users on the Internet. There will be data of intermediate sensitivity, which you want to be able to access while away from your home, but which you do not plan to share with others.
- ii. You require the ability to backup your data, including support for periodic off-site backup of data.
- iii. Your home network supports many “appliances” including security cameras, DVR systems such as Tivo, Televisions, Entertainment systems, and home automation systems capable of controlling lights and unlocking doors.
- iv. Your network supports multiple home computers, including tablets, smartphones, laptop computers, and desktop computers.
- v. You have a single connection into your network through a cable modem, DSL, or FiOS or similar capability, and you will deploy a router and wireless system for your network.

At this point, I could ask the single question, how will you secure this system, and you could write 200 pages and the question would be impossible for us to grade. As such, I can't ask such an open ended questions and instead ask a few specific questions which by no means cover the entire space of options.



# 2013 Final Exam – Q3.3

---

- a. In designing the network that will meet the requirements about file systems above, how will you protect the critically sensitive information differently than the other classes of information? How will you share the less sensitive information with other users on the internet? How might you support your own personal access to data of intermediate sensitivity, which you need to access when traveling? How will you protect the highly sensitive data, which needs to be readily accessible from your computers when you are at home? (10 points)
- b. I mentioned that your network will have a router, most likely at the point of connection to the internet, but which is also responsible for forwarding packets among the other devices on your network. Tell me what capabilities you will require on this device, in order to improve the security of your home network as a whole. Please be sure to note that while it will obviously have firewall functionality, there should be a lot more that it does too. (10 points)
- c. Defense in Depth – There will inevitably be security vulnerabilities on the devices in your home network. Group the devices into classes based on the impact of a device vulnerability on the security of the system as a whole, explain the impact, and describe how you can reduce the impact (or if aspects of your design above already reduce that impact, explain how it does so). (10 points)
- d. System Updates – With all the devices listed above, you are certain to require software updates for many of these devices. Discuss for which devices you are likely to enable automatic software updates, explain any vulnerabilities created by said choice, and how the impact of those vulnerabilities might be mitigated elsewhere in the system. Understand that for some of these devices, you will not be able to change the way updates are processed or validated, but can only enable or disable automatic updates. (10 points)

# **2020 Final Exam – Q1**

## **Match systems with Vulnerabilities**

---

- |   |                                   |
|---|-----------------------------------|
| 1. Smartcards                                       | a) Modification of returned data  |
| 2. Digital Signatures                               | b) Man in the Middle attack       |
| 3. Diffie Hellman Key Exchange                      | c) System or end-point Subversion |
| 4. The Domain Name System (traditional, NOT DNSSEC) | d) Worm                           |
| 5. Host Based Intrusion Detection                   | e) Stolen Credentials             |
| 6. Kerberos   | f) Phishing or password guessing  |
| 7. Host Based Firewalls                             |                                   |
| 8. Trusted Computing                                |                                   |

# **2020 Final Exam – Q2**

## **Short and Medium Answers**

---

- a. **Attestation – What is the meaning of attestation in trusted computing? How is attestation implemented / accomplished by the Trusted Platform Module (TPM). In answering the second part of this question, please note that there are multiple steps that occur at different times. Do not just describe the final step. (10 points)**
- b. **IPSec Authentication – Explain how authentication for IPSec in transport mode is fundamentally different from authentication of connections through HTTPS (SSL or TLS) and also how it is different from authentication performed by an application using a method like Kerberos. I am not concerned with the differences in the protocols used, but rather in the fundamental differences in what we know once the authentication steps are completed. (10 points)**
- c. **How is Secure DNS (i.e. DNSSEC) similar to public key infrastructure used by SSL and TLS. What entities or components in DNSSEC corresponded to the Certification Authority (CA) and to certificates in SSL/TLS. (10 points)**
- d. **List some of the advantages of a network-based intrusion detection system over a monolithic intrusion detection system located solely on the end-system that is being protected. (10 points)**

# **2020 Final Exam – Q3**

## **Impact of Pandemic on Security**

As a result of the pandemic, more and more employees (and students, and faculty) are working from home than ever before. This change in the location of our work creates significant changes to computer security technologies. Many of the assumptions we have made in the past no longer apply, and this changes the effectiveness of various security techniques and technologies. In this question you are asked to comment on some of these changes, and to suggest approaches to mitigate the impact these changes have on security.

- a. **Containment** – In the second lecture following the mid-term exam we discussed the placement of data in systems, and I used the term containment architecture to describe the relationship of the different protection domains in a system, and the placement of different kinds of data in those domains relative to the locations from which different classes of users required access.
- **Discuss how increased instances of work from home has changed the boundaries of the containment architecture for many organizations. Discuss also the technologies that are used to provide isolation / separation of protection domains both prior to the pandemic, and during the pandemic when more employees work from home.**
  - **Are there any organizational steps and guidelines (company policies) that could be applied to ensure that the containment architecture for the organization when employees work from home is as close as possible to that when employees worked from the office? (10 points)**

# **2020 Final Exam – Q3**

## **Impact of Pandemic on Security**

- b. Discuss some of the difficulties for corporate intrusion detection system when applied to systems running in the work-from-home configuration. (5 points)**
  
- c. Discuss the potential use of trusted computing technologies (including use of a Trusted Platform Module (TPM) to ensure that corporate information is only accessed and processed in accordance with company policy even when applications are running in an employee's home environment. Explain how your approach would prevent an adversary from accessing such data even though subversion (virus, trojan horses) of applications on the employee's computer system. (20 points)**