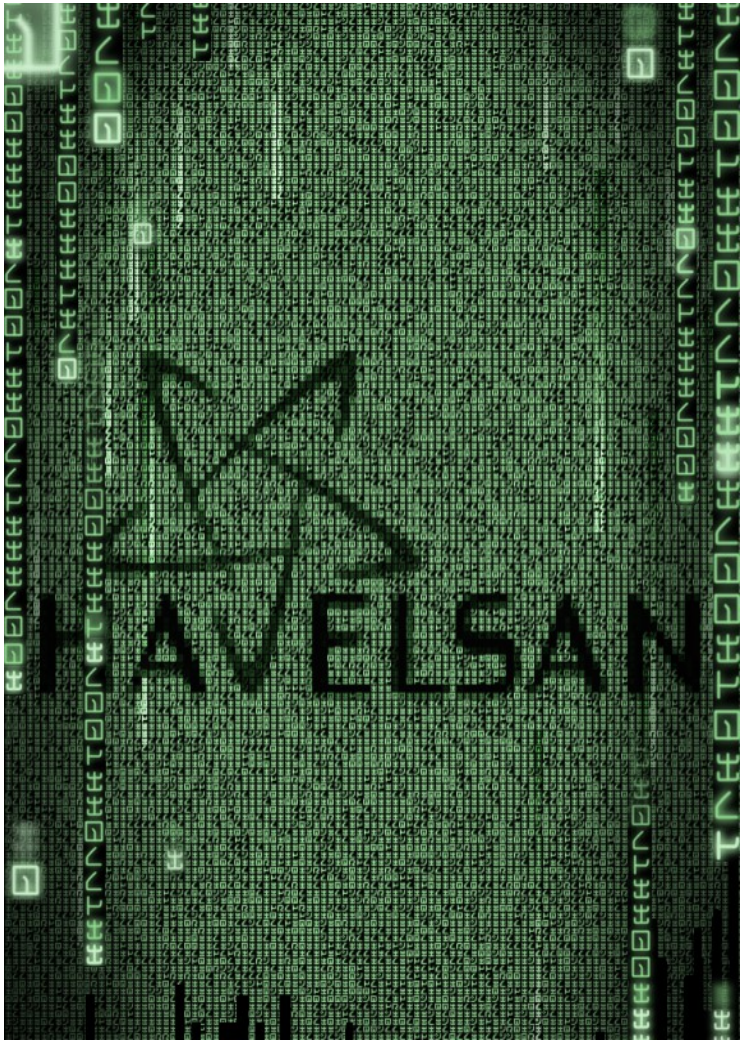
Bilkent University
CS 479

Basic Definitions of Cyber Security
Overview of Cyber Threats
Cryptography

Emre Yüce, PhD
HAVELSAN

Ankara, February 19

Course Outline



- Security Concepts
- Understanding Threats
- Security Design Principles
- Threats and Attacks
- Cryptography
- Corporate Security

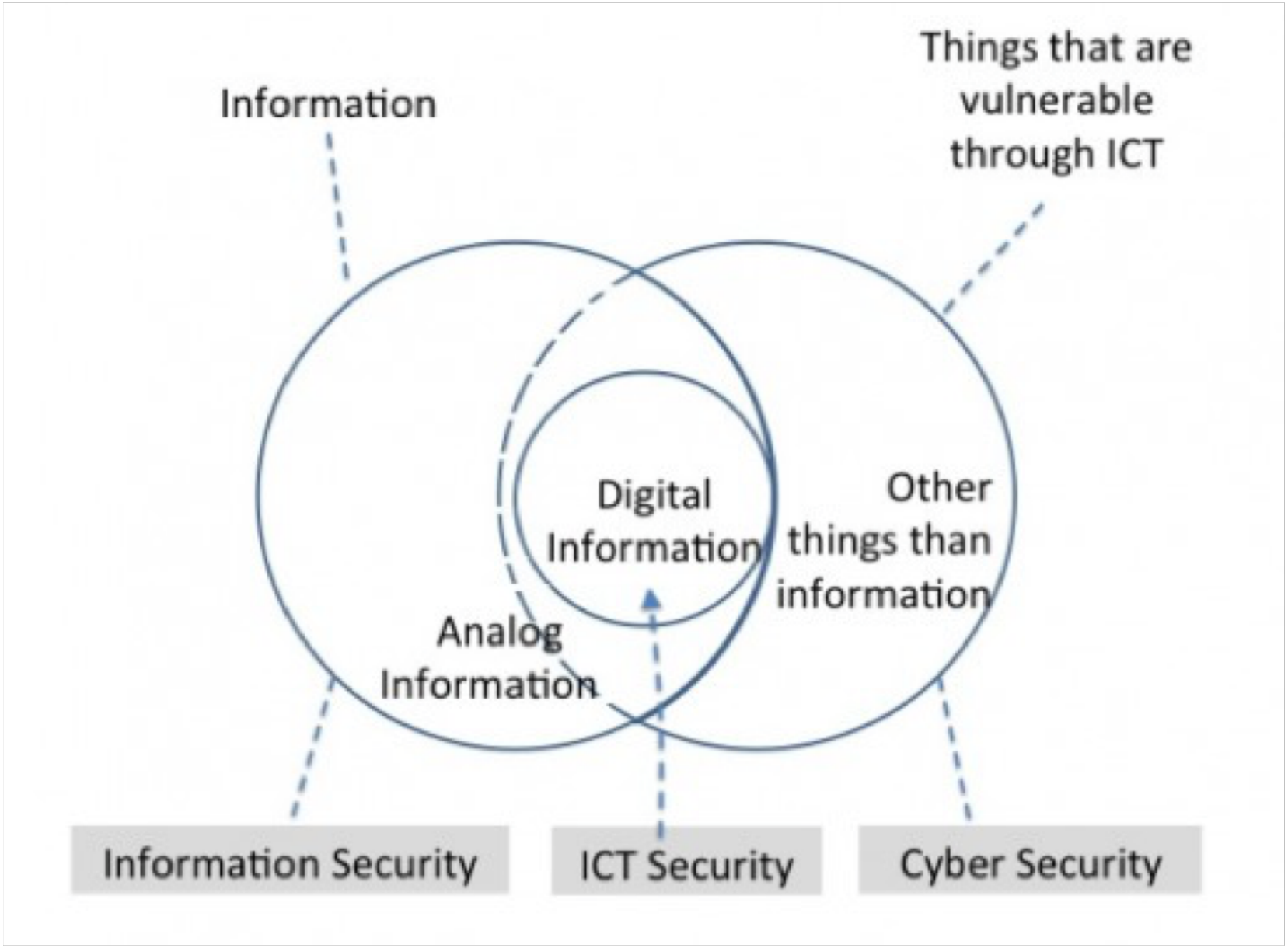
Information vs. Cyber Security

Cyber Security: The ability to protect or defend the use of cyberspace from cyber attacks.

Information Security: Protecting information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide —

1) **confidentiality**, which means preserving authorized restrictions on access and disclosure, including means for protecting personal privacy and proprietary information; and
2) **integrity**, which means guarding against improper information modification or destruction, and includes ensuring information non-repudiation and authenticity; 3) **availability**, which means ensuring timely and reliable access to and use of information.

Information vs. Cyber Security



Security Concepts

- Authentication
- Authorization
- Confidentiality
- Data / Message Integrity
- Accountability
- Availability
- Non-Repudiation

Authentication

- Identity Verification

- How can Bob be sure that he is communicating with Alice?

- Three General Ways:
  - Something you **know (i.e. Passwords)**
  - Something you **have (i.e., Tokens)**
  - Something you **are (i.e., Biometrics)**

Something you KNOW

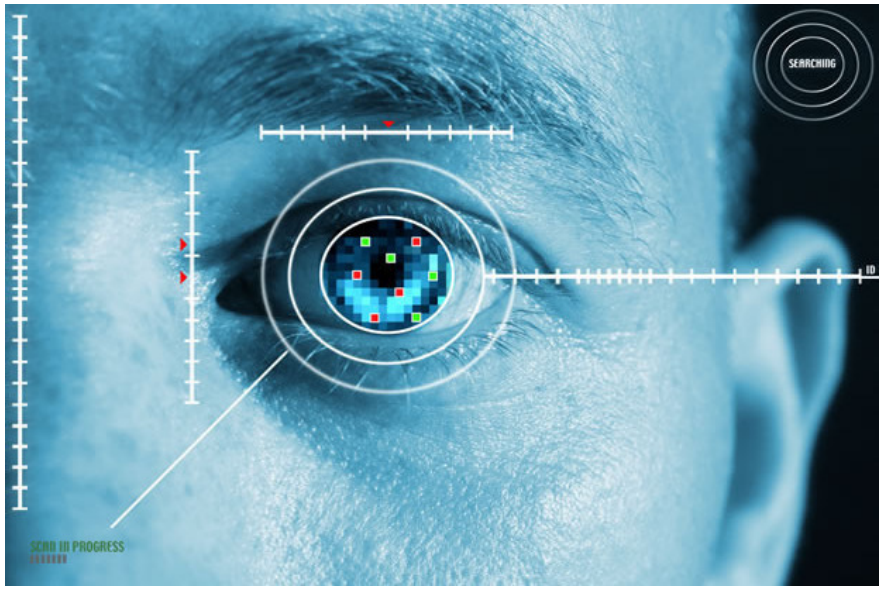
- Example: Passwords
  - **Pros**:
    - Simple to implement
    - Simple for users to understand
  - **Cons**:
    - Easy to crack (unless users choose strong ones)
    - Passwords are reused many times
- One-time Passwords (**OTP**): different password used each time, but it is difficult for user to remember all of them

Something you HAVE

- OTP Cards: generates new password each time user logs in
  - Internet Banking
- Smart Card: tamper-resistant, stores secret information, entered into a card-reader
  - New citizenship identity cards
- Token / Key
- ATM Card
- Strength of authentication depends on difficulty of forging

Something you ARE

- Biometrics
- Pros: “raises the bar”
- Cons: false negatives/positives, social acceptance, key management
  - false positive: authentic user rejected
  - false negative: impostor accepted



Remarks

- Two-factor Authentication: Methods can be combined (i.e. ATM card & PIN)

- Who is authenticating who?
  - Person-to-computer?
  - Computer-to-computer?

- Three types (e.g. SSL):
  - Client Authentication: server verifies client's id
  - Server Authentication: client verifies server's id
  - Mutual Authentication (Client & Server)

Authorization

- Checking whether a user has permission to do some action
- Who you are vs what you are allowed to do
- Is a “subject” (Alice) allowed to access an “object” (open a file)?
- **Access Control List**: mechanism used by many operating systems to determine whether users are authorized to conduct different actions

Access Control Lists (ACLs)

- Set of three-tuples
  - <User, Resource,Privilege>
  - Specifies which users are allowed to access which resources with which privileges

- Privileges can be assigned based on roles (e.g. admin)

Confidentiality

- Goal: Keep the contents of communication or data on storage secret
- Example: Alice and Bob want their communications to be secret from Eve

- Key – a secret shared between Alice & Bob
- Sometimes accomplished with
  - Cryptography, Steganography, Access Controls, Database Views

Message/Data Integrity

- Data Integrity = No Corruption
- **Man in the middle (MITM) attack**: Has Mallory tampered with the message that Alice sends to Bob?
- Integrity Check: Add redundancy to data/messages

Message/Data Integrity

- Techniques:
  - Hashing (MD5, SHA-1, ...), Checksums (CRC...)
  - Message Authentication Codes (MACs)

- Different From Confidentiality:
  - A-> B: “The value of x is 1” (not secret)
  - A -> M -> B: “The value of x is 10000” (BAD)
  - A -> M -> B: “The value of y is 1” (BAD)

ASSIGNMENT – 1

- Create a simple text report answering the following questions.

- What is the integrity check mechanism for
  - TR identity numbers?
  - credit card numbers?
- Write a script that generates
  - a random valid TR identity number.
  - a random valid credit card number.

- Send the outcomes to emre.yuce@bilkent.edu.tr
- **DEADLINE**: 13.02.2019 23:59

Accountability

- Able to determine the attacker or user
- Logging & Audit Trails
- Requirements:
  - Secure Timestamping (OS vs. Network)
  - Data integrity in logs & audit trails, must not be able to change trails, or be able to detect changes to logs
  - Otherwise attacker can cover their tracks.

Availability

- Uptime, Free Storage
  - Ex. Dial tone availability, System downtime limit, Web server response time

- Solutions:
  - Add redundancy to remove single point of failure
  - Impose “limits” that legitimate users can use

Availability

- Goal of **DoS (Denial of Service)** attacks are to reduce availability
  - Malware used to send excessive traffic to victim site
  - Overwhelmed servers can’t process legitimate traffic
  - **DDoS**

Non-Repudiation

- Undeniability of a transaction
- Alice wants to prove to Trent that she did communicate with Bob
- Generate evidence / receipts (digitally signed statements)
- Often not implemented in practice, credit-card companies become de facto third-party verifiers

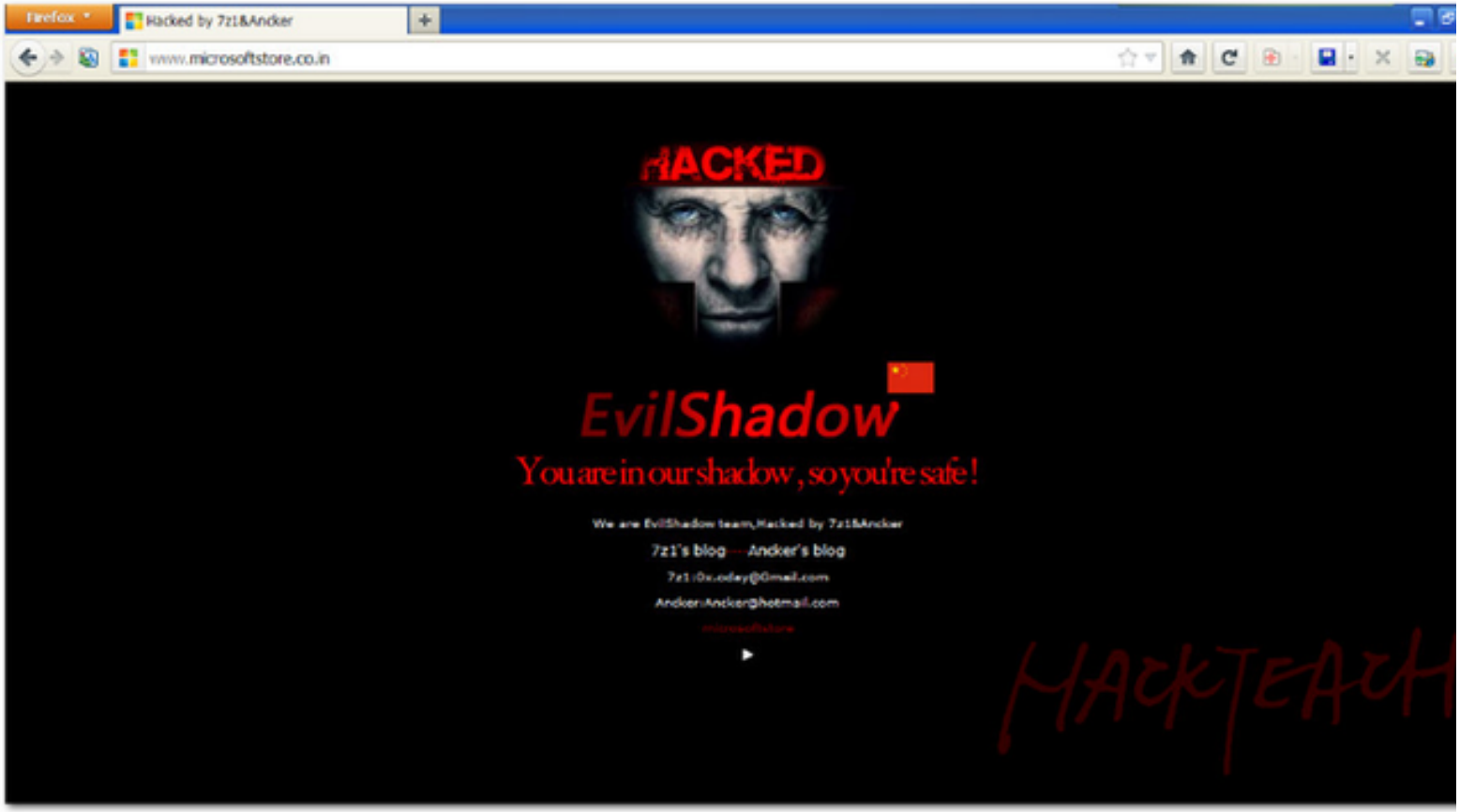
Understanding Threats

- ID & Mitigate Threats
  - Defacement
  - Infiltration
  - Phishing
  - Pharming
  - Spam
  - Insider Threats
  - Click Fraud
  - Denial of Service
  - Data Theft/Loss

Defacement

- Online Vandalism, attackers replace legitimate pages with illegitimate ones
- Targeted towards political web sites
- Ex: White House website defaced by anti-NATO activists, Chinese hackers

Defacement


Firefox
Hacked by 7z1&Ancker
www.microsoftstore.co.in
HACKED
EvilShadow
You are in our shadow, so you're safe!
We are EvilShadow team,Hacked by 7z1&Ancker
7z1's blog
Ancker's blog
HACKTEACH

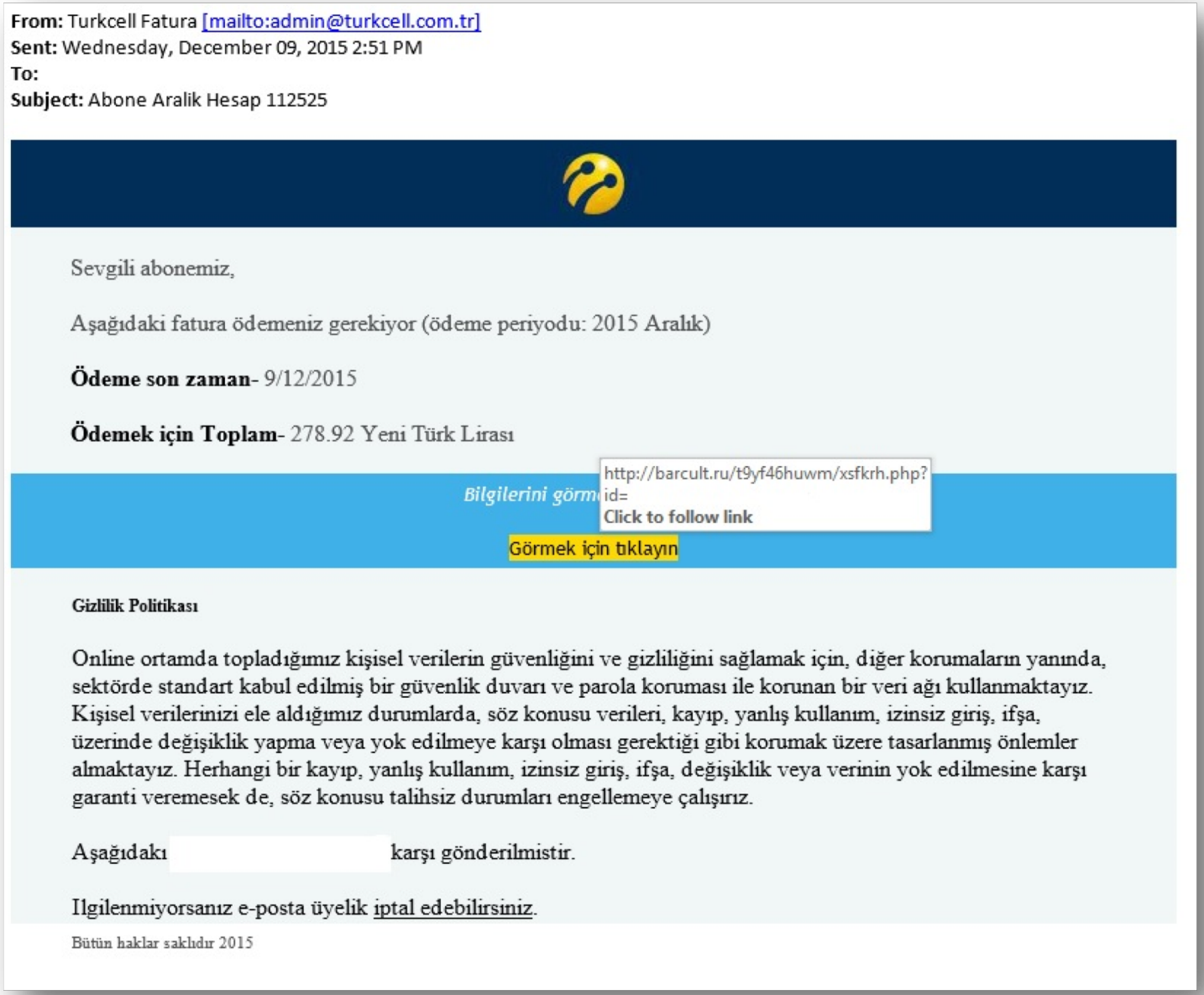
Infiltration

- Unauthorized parties gain access to resources of computer system (e.g. CPUs, disk, network bandwidth)
- Could gain read/write access to back-end DB
- Different goals for different organizations
  - Political site only needs integrity of data
  - Financial site needs integrity & confidentiality

Phishing

- Attacker sets up spoofed site that looks real
  - Attracts users to enter login credentials and stores them
  - Usually sent through an e-mail with link to spoofed site asking users to “verify” their account info
  - The links might be disguised through the click texts
  - Careful users can see actual URL if they hover over link
- Spear Phishing: Phishing attack targeting a specific person.

Phishing



From: Turkcell Fatura [mailto:admin@turkcell.com.tr]
Sent: Wednesday, December 09, 2015 2:51 PM
To:
Subject: Abone Aralik Hesap 112525

Sevgili abonemiz,

Aşağıdaki fatura ödemeniz gerekiyor (ödeme periyodu: 2015 Aralık)

Ödeme son zaman- 9/12/2015

Ödemek için Toplam- 278.92 Yeni Türk Lirası

Bilgilerini görm

http://barcult.ru/t9yf46huwm/xsfkrh.php?id=
Click to follow link

Görmek için tıklayın

Gizlilik Politikası

Online ortamda topladığımız kişisel verilerin güvenliğini ve gizliliğini sağlamak için, diğer korumaların yanında, sektörde standart kabul edilmiş bir güvenlik duvarı ve parola koruması ile korunan bir veri ağı kullanmaktayız. Kişisel verilerinizi ele aldığımız durumlarda, söz konusu verileri, kayıp, yanlış kullanım, izinsiz giriş, ifşa, üzerinde değişiklik yapma veya yok edilmeye karşı olması gerektiği gibi korumak üzere tasarlanmış önlemler almaktayız. Herhangi bir kayıp, yanlış kullanım, izinsiz giriş, ifşa, değişiklik veya verinin yok edilmesine karşı garanti veremesek de, söz konusu talihsiz durumları engellemeye çalışırız.

Aşağıdaki karşı gönderilmistir.

Ilgilenmiyorsanız e-posta üyelik iptal edebilirsiniz.

Bütün haklar saklıdır 2015

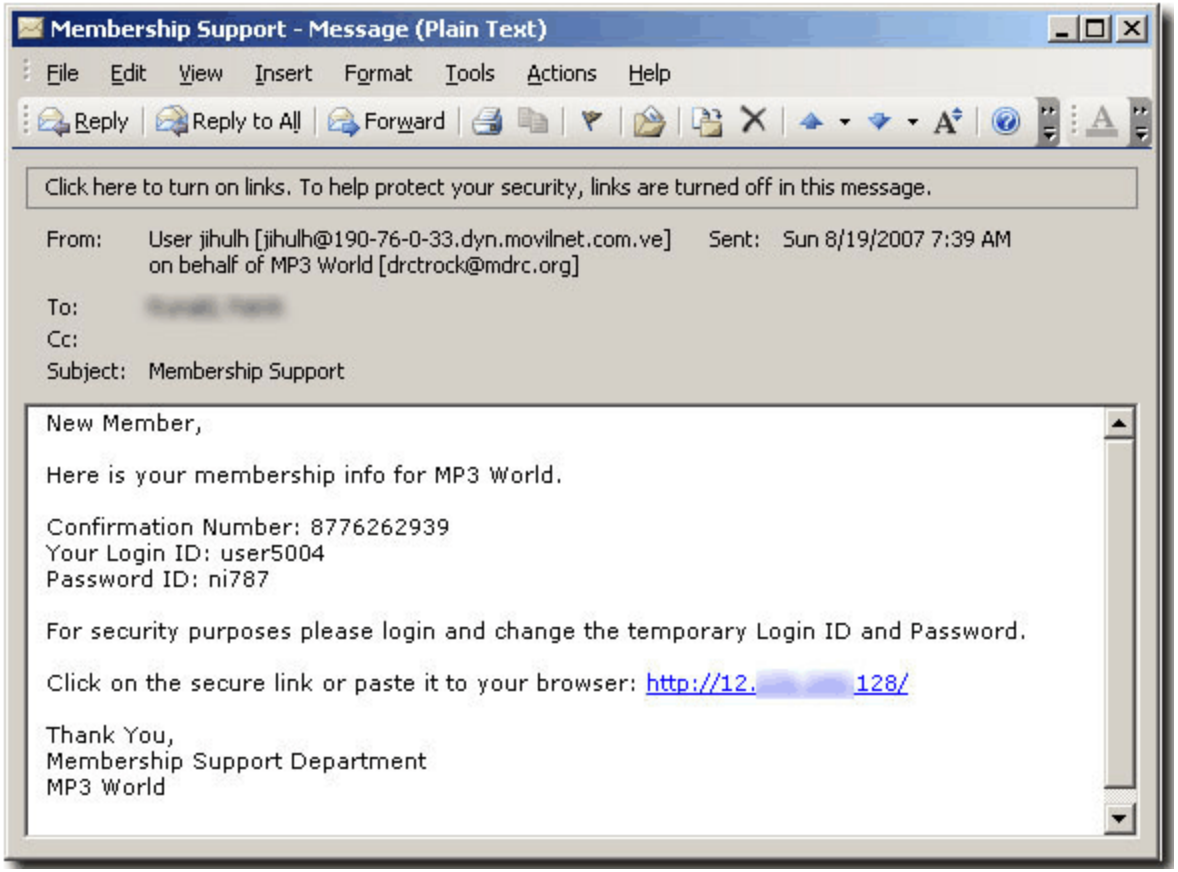
Pharming

- Like phishing, attacker's goal is to get user to enter sensitive data into spoofed website
- DNS Cache Poisoning – attacker is able to redirect legitimate URL to their spoofed site
- DNS translates URL to appropriate IP address
- Attacker makes DNS translate legitimate URL to their IP address instead and the result gets cached, poisoning future replies as well

Spam

- Sending unsolicited messages (spam), especially advertising, as well as sending messages repeatedly on the same site.
- 120 billion spam emails per day world-wide
- 77% of emails are spam on average
- Modern delivery vehicle for email attachments

Spam


Membership Support - Message (Plain Text)
File Edit View Insert Format Tools Actions Help
Reply Reply to All Forward
Click here to turn on links. To help protect your security, links are turned off in this message.
From: User jihulh [jihulh@190-76-0-33.dyn.movilnet.com.ve] Sent: Sun 8/19/2007 7:39 AM
on behalf of MP3 World [drctrock@mdrc.org]
To:
Cc:
Subject: Membership Support
New Member,
Here is your membership info for MP3 World.
Confirmation Number: 8776262939
Your Login ID: user5004
Password ID: ni787
For security purposes please login and change the temporary Login ID and Password.
Click on the secure link or paste it to your browser: http://12. 128/
Thank You,
Membership Support Department
MP3 World

Example: Storm worm

- Discovered January 2007

- Backdoor Trojan horse
  - email that reports a storm in Europe
  - has executable attachment
  - opens infected host to remote control

- World's most powerful supercomputer
  - peer-to-peer botnet

Storm E-mail Subjects

- A killer at 11, he's free at 21 and kill again
- U.S. Secretary of State Condoleezza Rice has kicked German Chancellor Angela Merkel
- British Muslims Genocide
- Naked teens attack home director
- Radical Muslim drinking enemies' blood
- Chinese/Russian missile shot down Russian/Chinese satellite/aircraft
- Saddam Hussein safe and sound!
- Venezuelan leader: "Let's the War beginning"
- Fidel Castro dead
- FBI vs. Facebook

Storm worm behaviour

- Patience:
  - Has active and inactive periods
- Separation of duties:
  - Small number of hosts spread worm further
  - Smaller number of hosts serve as control
  - Large number of hosts wait for tasks
- No damage, little impact on host
- Constant change
  - Delivery mechanism, payload
  - DNS manipulations: “fast flux”

Storm Worm Summary

- Scale: 1 to 10 million infected hosts
  - Billions of spam per day
  - Estimate: running at only 10-20% of capacity
- Rumors of being leased to criminal groups
- Who controls storm ?

Insider Threats

- Attacks carried out with cooperation of insiders
  - Insiders could have access to data and leak it
  - Ex: DB and Sys Admins usually get complete access

- Separation of Privilege / Least Privilege Principle
  - Provide individuals with only enough privileges needed to complete their tasks
  - Don’t give unrestricted access to all data and resources

Insider Threats

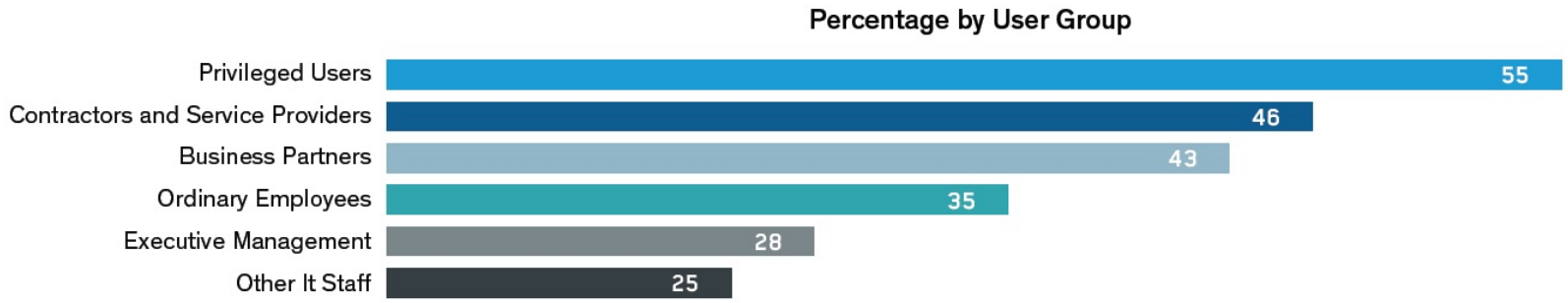


Figure 1: The global position for insiders who pose the largest risk to an organization

Source: 2015 Vormetric Insider Threat Report

Click Fraud

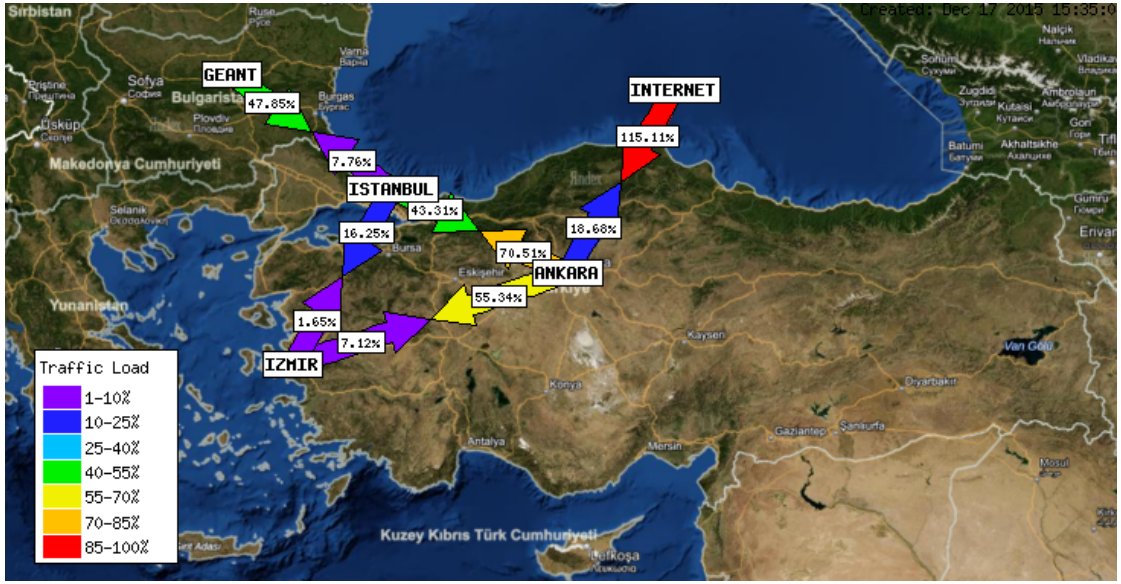
- Targeted against pay-per-click ads
- Attacker could click on competitor's ads
  - Depletes other's ad budgets, gains exclusive attention of legitimate users
- Site publishers could click on ads to get revenue
- Automated through malware such as botnets

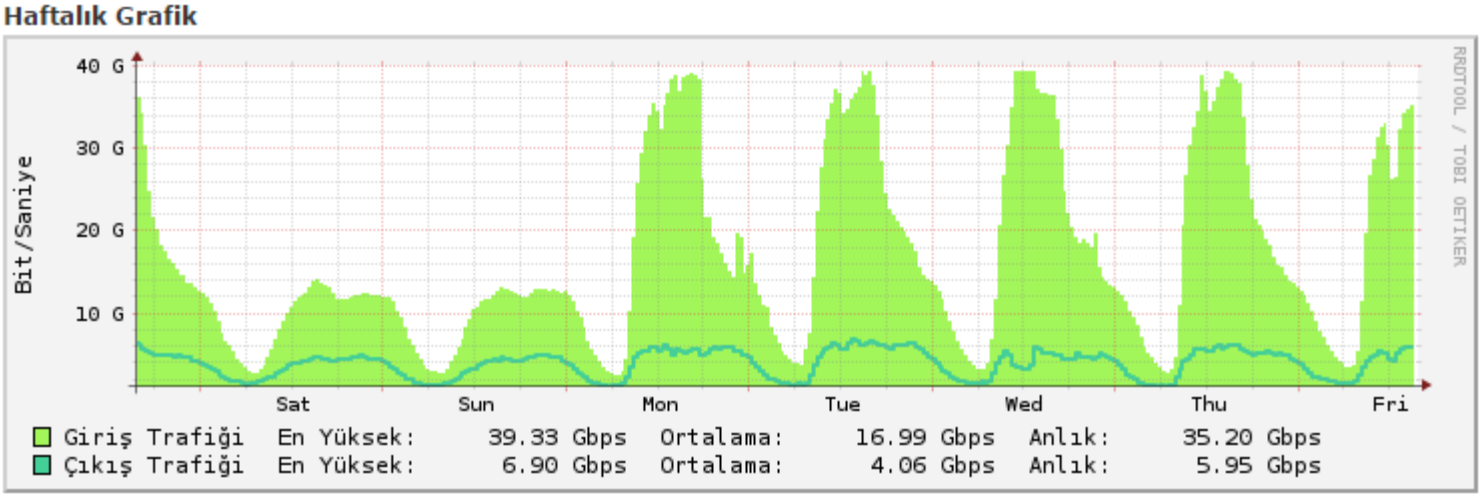
Denial of Service (DoS)

- Attacker floods server with packets causing it to drop legitimate packets
  - Makes service unavailable, downtime = lost revenue
- Particularly a threat for financial and e-commerce vendors
- Can be automated through botnets
- Distributed Denial of Service (DDoS)

DDoS attack on NIC.TR

- December 2015
- >50Gbps traffic
- ULAKNET
- RIPE





DDoS attack on BBC

- A group calling itself **New World Hacking** said that the attack reached **602Gbps**. (Jan 2016)
- It seems that **New World Hacking** may be affiliated with an online DDoS tool called **BangStresser**, which delivers attacks as a service.
- In 2015, a similar group, the **Lizard Squad**, conducted a marketing campaign for their DDoS service, the **Lizard Stressor**.
- New World Hacking is claiming to be using **Amazon servers** to generate actual attack bandwidth.

Data Theft and Data Loss

- Several Examples: BofA, ChoicePoint, VA
  - BofA: backup data tapes lost in transit
  - ChoicePoint: fraudsters queried DB for sensitive info
  - VA: employee took computer with personal info home & his home was burglarized

- Even for encrypted data, should store key in separate media

Threat Modeling

Application Type	Threat
Governmental web sites	Defacement
Electronic commerce	Compromise accounts, Denial-of-Service
Military institution	Infiltration, access to classified data

Security Design

- Design features with security in mind
  - Not as an afterthought
  - Hard to “add-on” security later

- Define concrete, measurable security goals. Ex:
  - Only certain users should be able to do X. Log action.
  - Output of feature Y should be encrypted.
  - Feature Z should be available 99.9% of the time

- Bad Examples: Windows 98, Internet

Windows 98

- Diagnostic Mode:
  - Accessed through 'F8'
  - Can bypass password key when booting protections, giving attacker complete access to hard disks & data

- Username/Password Security was added as an afterthought

- Should have been included at the start, then required it for entering diagnostic mode

Windows 7

- Sticky keys bug!
- Physical access is dangerous!

https://www.youtube.com/watch?v=Y6BeG5LjJ9g

The Internet

- All nodes originally university or military (i.e.
- trusted) since it grew out of DARPA

- With commercialization, lots of new hosts, all allowed to connect to existing hosts regardless of whether they were trusted

- Deployed Firewalls: allows host to only let in trusted traffic
- Loopholes: lying about IPs, using cleared ports

The Internet

"On the Internet, nobody knows you're a dog."

IP Whitelisting & Spoofing

- IP Whitelisting: accepting communications only from hosts with certain IP addresses
- IP Spoofing attack: attacker mislabels (i.e. lies) source address on packets, slips past firewall
- Response to spoofing sent to host, not attacker
  - Multiple communication rounds makes attack harder
  - May DoS against legitimate host to prevent response

IP Spoofing & Nonces

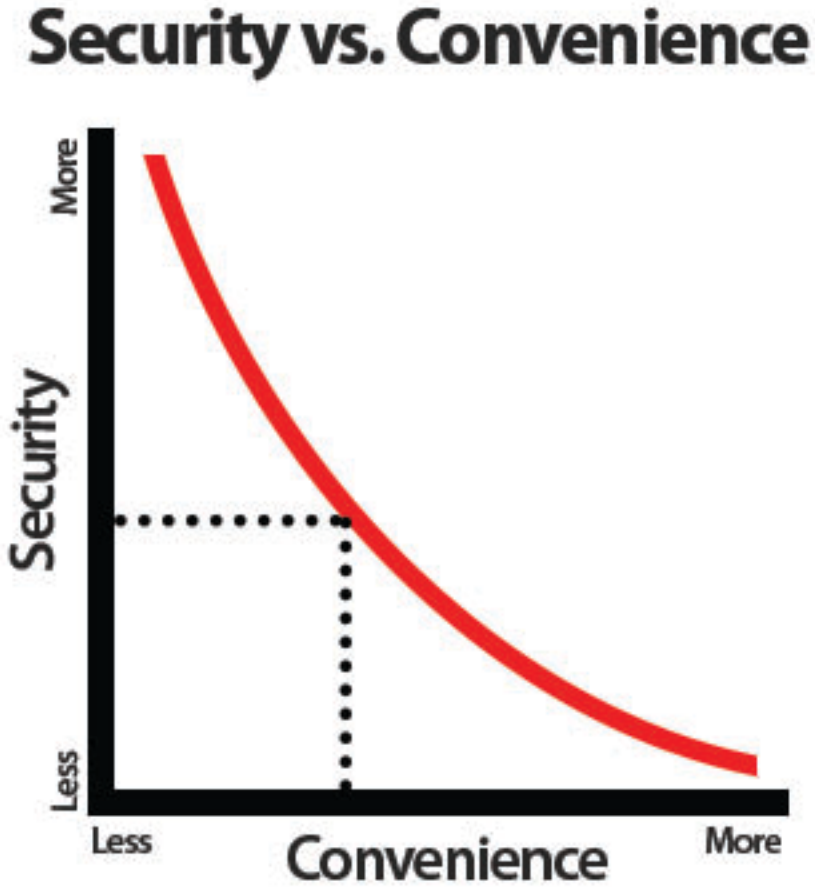
- Nonce: one-time pseudo-random number

- Attaching a nonce to a reply and requesting it to be echoed back can guard against IP spoofing
- Attacker won't know what reply to fake

- Spoofing easier for non-connection-oriented protocols (e.g. UDP) than connection-oriented (e.g. TCP)

- TCP sequence #s should be random, o/w attacker can predict and inject packets into conversation

Convenience and Security



Convenience and Security

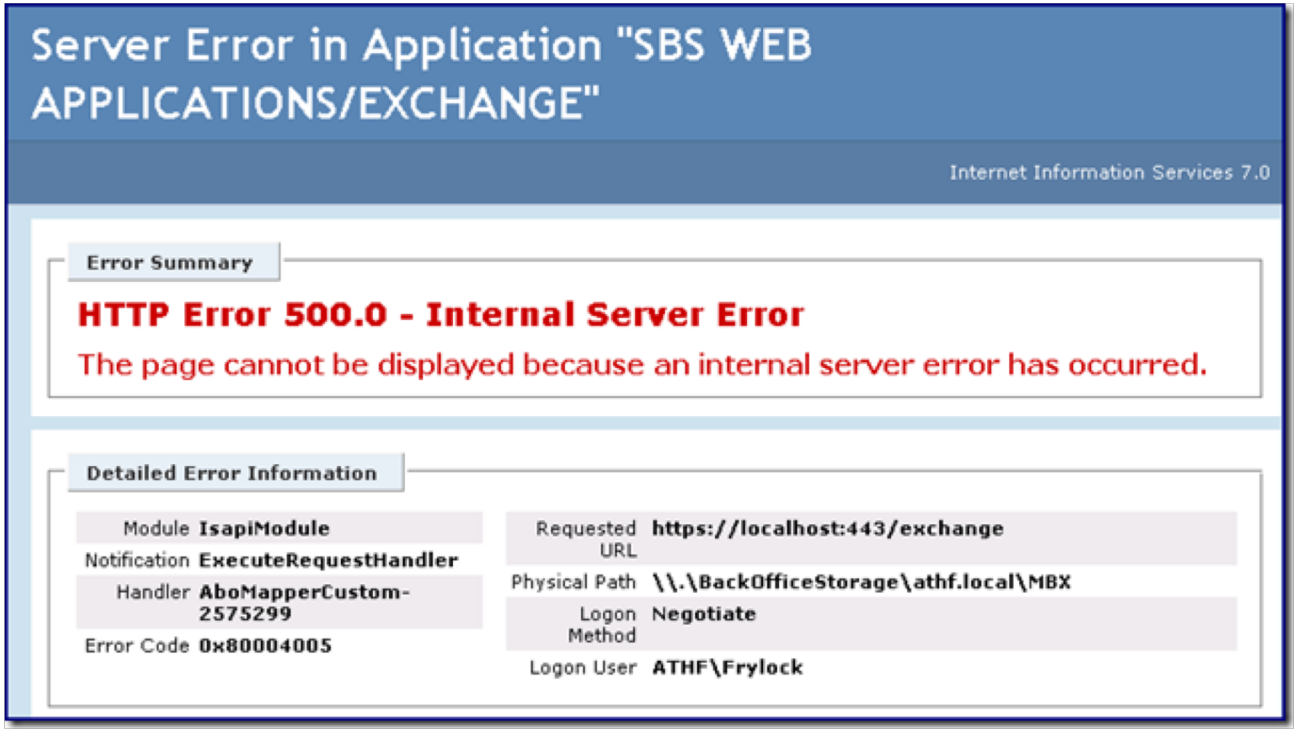
- Sometimes inversely proportional
  - More secure → Less convenient
  - Too Convenient → Less secure

- If too inconvenient → unusable → users will workaround → insecure
  - Ex: users may write down passwords

- Good technologies increase both: relative security benefit at only slight inconvenience

Security in Software Requirements

- Robust, consistent error handling
- Handle internal errors securely – don't provide error messages to potential attackers!
- Validation and Fraud Checks



Security in Software Requirements

- Both dev & testers should get requirements
- Should have test cases for security too: Does it malfunction when provided bad input?
- **Ping-of-Death**: sending a packet of data can cause server to crash
  - Ex: DoS attack on SimpleWebServer
  - Ex: Nokia GGSN crashes on packet with TCP option field set to 0xFF

Handling Internal Errors Securely

- Error messages and observable behavior can tip off an attacker to vulnerabilities
- Fault Injection: Providing a program with input that it does not expect and observing its behavior
- “Ethical” hackers often hired to find such bugs

Best Practice

- Access Control Security: Only certain users can do certain functions

- Auditing: Maintain log of users' sensitive actions
- Confidentiality: encrypt certain functions' output
- Availability: Certain features should be available almost always

- Include these requirements in design docs!

Security by Obscurity

- Trying to be secure by hiding how systems and products work (to prevent info from being used by attacker)
- Ex: Military uses Need to Know basis
- Maybe necessary, but not sufficient to prevent determined attackers

Flaws in the Approach

- What assumptions to make about adversary?
  - Knows algorithms? Or not?
  - Algorithms in “binary” secret?

- Attackers can probe for weaknesses
  - Reverse engineeing
  - Observe behavior in normal vs. aberrant conditions (use fault injection)
  - Fuzzing: systematically trying different input strings to find an exploit
  - Blackmail insiders

Best practice

- Kerckhoffs ’ doctrine (1883): “The method used to encipher data is known to the opponent, and that security must lie in the choice of key.”
  - Assume the worst case!
  - Obscurity alone is not sufficient.

- Compromised key can be changed without re-designing the system.
- Key is easier to be kept secret.

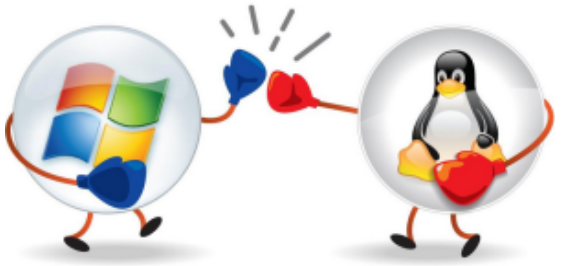
Things to Avoid

- Don’t invent your own encryption algorithm!

- Don’t embed keys in software!
- Nor in Windows Registry which is readable by all.

- Don’t Forget Code Reuse: reuse well-tested software known to be reliably secure instead of doing same thing from scratch.

Open vs. Closed Source



- Is open-source software secure?

- Open:
  - Some people might look at security of your application (if they care)
  - may or may not tell you what they find
- Closed:
  - Not making code available does not hide much
  - need diverse security-aware code reviews

- A business decision: Not a security one!

Secure Design Principles

Principle of Least Privilege

- Just enough authority to get the job done.

- Common world ex: Valet Keys
  - Valets can only start car and drive to parking lot

- Highly elevated privileges unnecessary
  - Ex: valet key shouldn't open glove compartment
  - Web server Ex: can read, not modify, html file
  - Attacker gets more power, system more vulnerable

Defense-in-Depth

- Also called redundancy/diversity: layers of defense, don’t rely on any one for security

- Examples
  - Banks: Security Guards, Bullet-Proof, Teller Window, Dye on Money
  - Many different types of magic and many levels of defense protecting the Sorcerer's Stone in Harry Potter

Prevent, Detect, Contain, and Recover

- Should have mechanisms for
  - preventing attacks,
  - detecting breaches,
  - containing attacks in progress,
  - and recovering from them.

- Detection particularly important for network security since it may not be clear when an attack is occurring.

Containment and Recovery

- Preventive techniques are not perfect; treat malicious traffic as a fact, not an exceptional condition.

- Should have containment procedures planned out in advance to mitigate damage of an attack that escapes preventive measures.
  - Design, practice, and test containment plan
  - Ex: If a thief removes a painting at a museum, the gallery is locked down to trap him.

Password Security

- Sys Admins can require users to choose strong passwords to prevent guessing attacks
- To detect, can monitor server logs for large # of failed logins coming from an IP address and mark it as suspicious
- Contain by denying logins from suspicious IPs or require additional checks (e.g. cookies)
- To recover, monitor accounts that may have been hacked, deny suspicious transactions

Diversity-in-Defense

- Using multiple heterogeneous systems that do the same thing
  - Use variety of OSes to defend against virus attacks.
  - Second firewall (different vendor) between server & DB.

- Cost: IT staff need to be experts in and apply to patches for many technologies.
  - Weigh extra security against extra overhead.

Securing the Weakest Link

- Information System is only as strong as its weakest link.

- Common Weak Links:
  - Weak Passwords: easy to crack
  - People: Social Engineering Attacks
  - Buffer Overflows from garbage input

Weak Passwords

- One third of users choose a password that could be found in the dictionary.
- Attacker can employ a dictionary attack and will eventually succeed in guessing someone's password.
- By using Least Privilege, can at least mitigate damage from compromised accounts.

Top Passwords

Worst passwords of the last five years					
	2015	**2014**	**2013**	**2012**	**2011**
#1	123456	123456	123456	password	password
#2	password	password	password	123456	123456
#3	12345678	12345	12345678	12345678	12345678
#4	qwerty	12345678	qwerty	abc123	qwerty
#5	12345	qwerty	abc123	qwerty	abc123
#6	123456789	1234567890	123456789	monkey	monkey
#7	football	1234	111111	letmein	1234567
#8	1234	baseball	1234567	dragon	letmein
#9	1234567	dragon	iloveyou	111111	trustno1
#10	baseball	football	adobe123	baseball	dragon
#11	welcome	1234567	123123	iloveyou	baseball
#12	1234567890	monkey	admin	trustno1	111111
#13	abc123	letmein	1234567890	1234567	iloveyou
#14	111111	abc123	letmein	sunshine	master
#15	1qaz2wsx	111111	photoshop	master	sunshine
#16	dragon	mustang	1234	123123	ashley
#17	master	access	monkey	welcome	bailey
#18	monkey	shadow	shadow	shadow	passw0rd
#19	letmein	master	sunshine	ashley	shadow
#20	login	michael	12345	football	123123
#21	princess	superman	password1	jesus	654321
#22	qwertyuiop	696969	princess	michael	superman
#23	solo	123123	azerty	ninja	qazwsx
#24	passw0rd	batman	trustno1	mustang	michael
#25	starwars	trustno1	000000	password1	football

Weak Passwords – Best Practice

- Use passwords of eight characters or more with mixed types of characters.
- Avoid using the same username/password combination for multiple websites.
- How many passwords do you have?
- Make the password more complex as the importance increases!

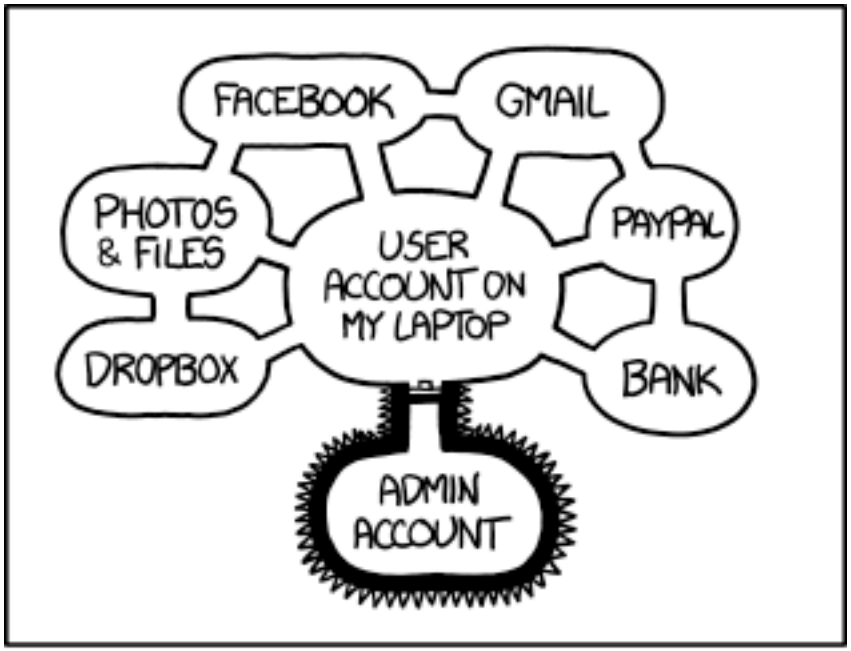
People

- Employees could fall for phishing attacks (e.g. someone calls them pretending to be the "sysadmin" and asks for their password).
  - Especially a problem for larger companies.

- Malicious Programmers
  - Can put back doors into their programs.
  - Should employ code review.

- Keep employees happy, less incentive for them to defraud company.
  - Also distribute info on need-to-know basis, perform background checks on hires.

Do you use logout button often?



IF SOMEONE STEALS MY LAPTOP WHILE I'M LOGGED IN, THEY CAN READ MY EMAIL, TAKE MY MONEY, AND IMPERSONATE ME TO MY FRIENDS,

BUT AT LEAST THEY CAN'T INSTALL DRIVERS WITHOUT MY PERMISSION.

Source: https://xkcd.com/1200/

Vulnerabilities Caused by Implementation Issues

- Correct Design can have bugs in implementation.
- Misuse of encryption can allow attacker to bypass it and access protected data.
- Inadvertent mixing of control and data.
  - Attacker feeds input data that's interpreted as a command to hijack control of program.
  - Ex: buffer overflows, SQL injection.

Fail-Safe

- Expect & Plan for System Failure

- Common world example: Elevators.
  - Designed with expectation of power failure.
  - In power outage, can grab onto cables or guide rails.

- Ex: If firewall fails, let no traffic in (fail-closed).
  - Deny access by default.
  - Don’t accept all (including malicious), because that gives attacker additional incentive to cause failure.

What have you learnt so far ?

- Cyber security - information security

- Confidentiality
- Integrity
- Availability

- Authentication
- Authorization
- Accountability
- Non-repudiation

- Defacement
- Infiltration
- Phishing
- Pharming
- Spam
- Insider threats
- Click fraud
- DoS & DDoS
- Data theft & data loss

- IP whitelisting - Spoofing

- Security by obscurity
- Kerckhoff's doctrine

- Principle of least privilege
- Defense in depth
- Diversity in defense
- Securing the weakest link

- Password security

- Fail-safe

Extra !!!
- DHCP
- DNS
- SSL/TLS
- Symmetric/Aymmetric encryption

Simplicity

- Security holes likely in complex software.
- Simpler design is easier to understand and audit.
- Choke point: centralized piece of code through which all control must pass.
  - Keeps security checks localized, easier to test.
- Less functionality = Less security exposure.

Usability

- Usable = users can easily accomplish the tasks they need to do with the software.

- Don’t rely on documentation: enable security features by default, design to be easy to use
  - Difficulty is in tradeoff with user convenience.

- Users are lazy (They ignore security dialogs)
  - Prevent users from committing insecure actions, assist them in doing it securely.

Usability for Security

- Security software is usable if the people who are expected to use it:
  - are reliably made aware of security tasks they need to perform.
  - are able to figure out how to successfully perform those tasks.
  - do not make critical errors.
  - are sufficiently comfortable with the interface to continue using it.

Security Features Do Not Imply Security

- Using one or more security algorithms/protocols will not solve all your problems!
  - Using encryption doesn't protect against weak passwords.
  - Using SSL doesn't protect against buffer overflows.

- Schneier: "Security is a process, not a product!"
  - Can never be completely secure, just provide a risk assessment (more testing lessening risk).
  - Attacker only needs to find one flaw, designers have to try and cover all possible flaws.
  - Security features can help, but can't stop bugs.

Security Principles Summary

- Employ a few key design principles to make system more secure.
  - Avoid elevated privileges.
  - Use layered defense (prevention, detection, containment, and recovery).
  - Secure weakest links.
  - Have fail-safes, i.e. crash gracefully.
  - Don’t enable unnecessary features.
  - Keep design simple, usable.
  - Security features can’t compensate for bugs.

Threats and Attacks

Threats and Attacks

- Malware
  - Virus,worm, rootkit, botnet, spyware and more.
- Buffer overflow
- SQL Injection

Virus & Worm

- Virus: program that copies itself into other programs.
  - Could be transferred through infected.
  - Rate dependent on human use disks.

- Worm: a virus that uses the network to copy itself onto other computers.

- Worms propagate faster than viruses.
  - Large # of computers to infect.
  - Connecting is fast (milliseconds).

History of Worms

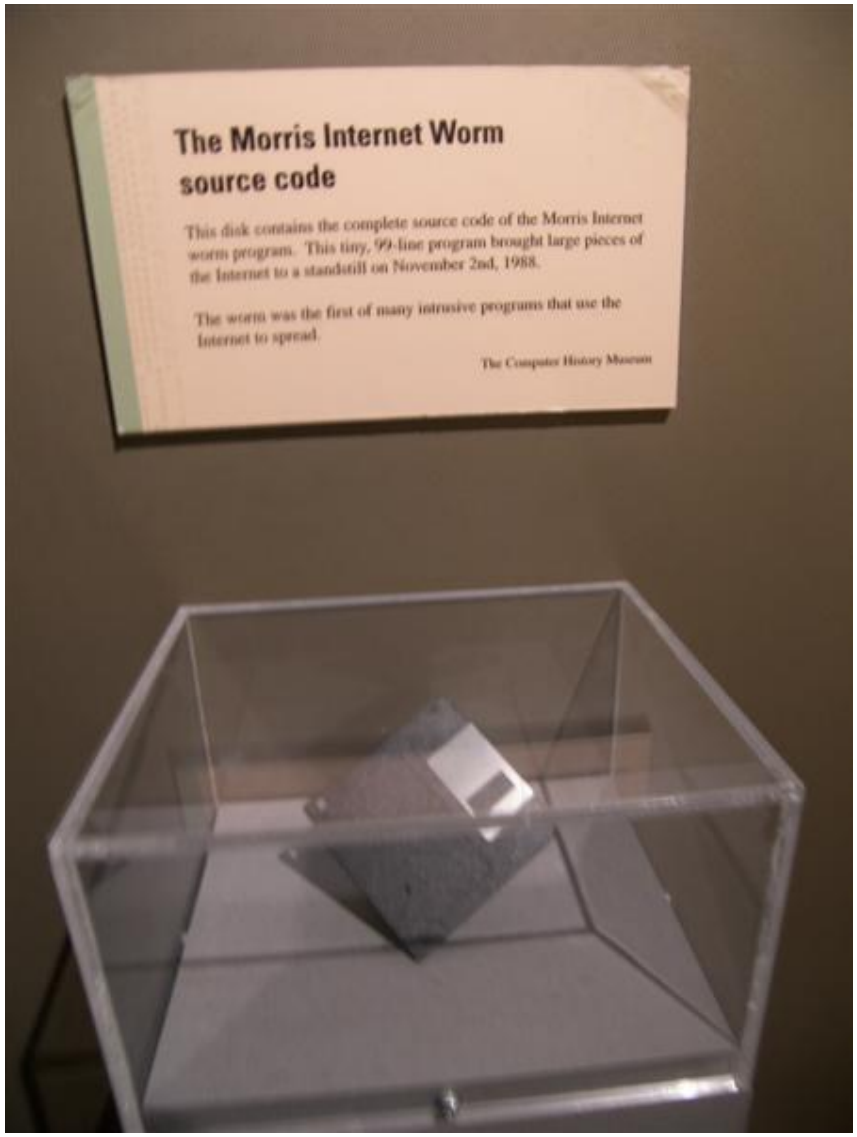
- Examples of how worms affect operation of entire Internet.

- First Worm: Morris Worm (1988).
- Code Red & Nimda (2001).
- SQL Slammer (2003).

Morris Worm: What It Did

- Damage: 6000 computers in just few hours
- Extensive network traffic by worm propagating
- What: just copied itself; didn't touch data
- Exploited and used:
  - Buffer overflow in fingerd (UNIX)
  - sendmail debug mode (execute arbitrary bit commands such as copying worm to another machine)
  - dictionary of 432 frequently used passwords to login and remotely execute commands via rexec, rsh

Morris Worm


The Morris Internet Worm
source code
This disk contains the complete source code of the Morris Internet worm program. This tiny, 99-line program brought large pieces of the Internet to a standstill on November 2nd, 1988.
The worm was the first of many intrusive programs that use the Internet to spread.
The Computer History Museum

The Morris Worm: What We Learned

- Diversity is good: Homogenity of OSes on network -> attacker can exploit vulnerabilities common to most machines

- Large programs more vulnerable to attack
  - Sendmail was large, more bug-prone
  - fingerd was small, but still buggy

- Limiting features limit holes: sendmail debug feature should have been turned off
- Users should choose good passwords: dictionary attack would have been harder

The Creation of CERT

- Computer Emergency Response Team (CERT) created due to damage and disruption caused by Morris worm
- Has become a leading center on worm activity and software vulnerability announcements
- Raises awareness about cyber-security

Other Types of Malware

- **Rootkits**: Imposter OS tools used by attacker to hide tracks.

- **Botnets**: Network of software robots attacker uses to control many machines at once to launch attacks (e.g. DDoS through packet click fraud) flooding.

- **Spyware**: Software that monitors activity of a system or its users without their consent.

Other Types of Malware

- **Keyloggers**: Spyware that monitors user keyboard or mouse input, used to steal usernames, passwords, credit card #s, etc...
- **Trojan Horses**: Software performs additional or different functions than advertised
- **Adware**: Shows ads to users w/o their consent
- **Clickbots**: Bot that clicks on ads, leads to click fraud (against cost-per-click or CPC ad models)

Distributing Malware

- Most malware distribution through drive-by downloads (i.e. automatic installation of binary when visiting website).
  - Uses pull-based model (e.g. links).
  - Maximizes exposure by getting as many links as possible to malware distribution site.

- Search engines such as Google mark pages as potentially malicious to prevent.

- Browsers use URL blacklists.

Zeus Botnet

- Spread via drive-by-downloads drive by downloads and phishing.
- First identified July 2007.
- Compromised over 74K FTP accounts in June 2009.
- Affected: Bank of America, NASA, Monster, ABC, Oracle, Cisco, Amazon, and BusinessWeek.

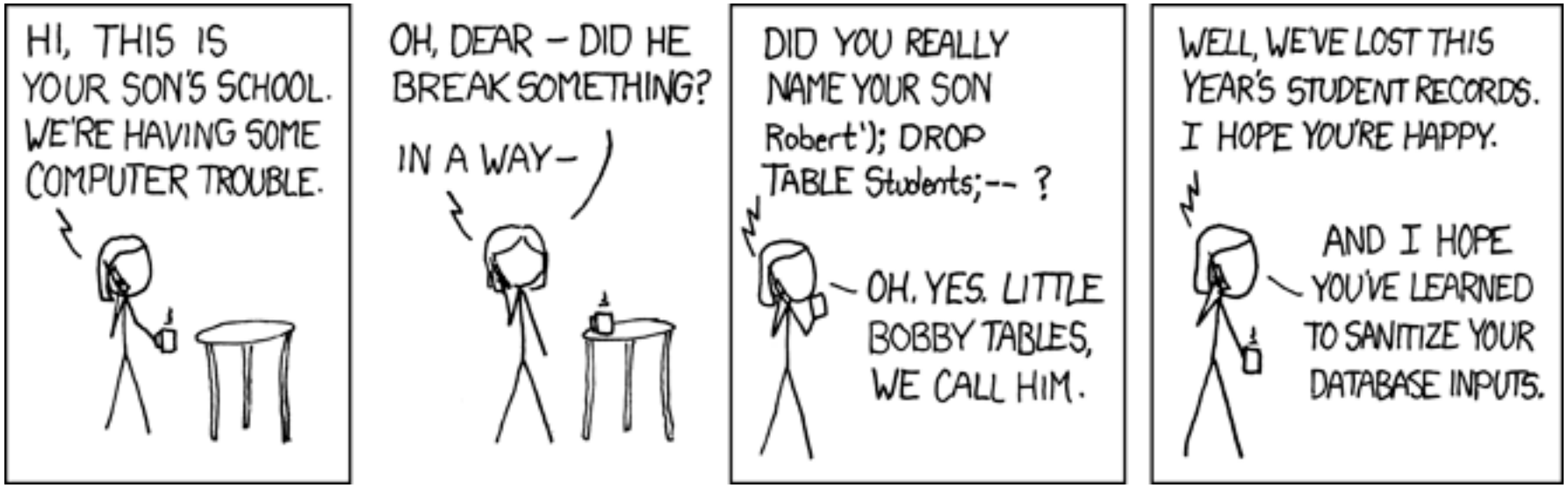
How a drive-by-download attack works

1. Inject legitimate web page with malicious code (e.g., JavaScript, IFRAME, etc) OR direct user to infected web page (e.g. Fake anti-virus or phishing).
2. Invoke client-side vulnerability (e.g., IE zero-day, PDF exploit, etc) OR use social engineering.
3. Deliver shellcode to take control.
4. Send “downloader” & deliver malware of attackers choice.

Protection Against Malware

- Malware exploits common vulnerabilities to spread + achieve widespread damage.
- Prevention
  - Eliminate Buffer Overflows (Programmers).
  - Don’t open email attachments (Users, SAs).
  - Disable unnecessary functionality (Users, SAs).
  - Use a secure browser (Users, , SAs).
  - Patch systems regularly (SAs).
- Detection
  - Update scanners with latest definitions.
  - Use auto-updating scanners when possible.
  - Employ programs such as Tripwire (SAs).

SQL Injection



Source: http://xkcd.com/327/

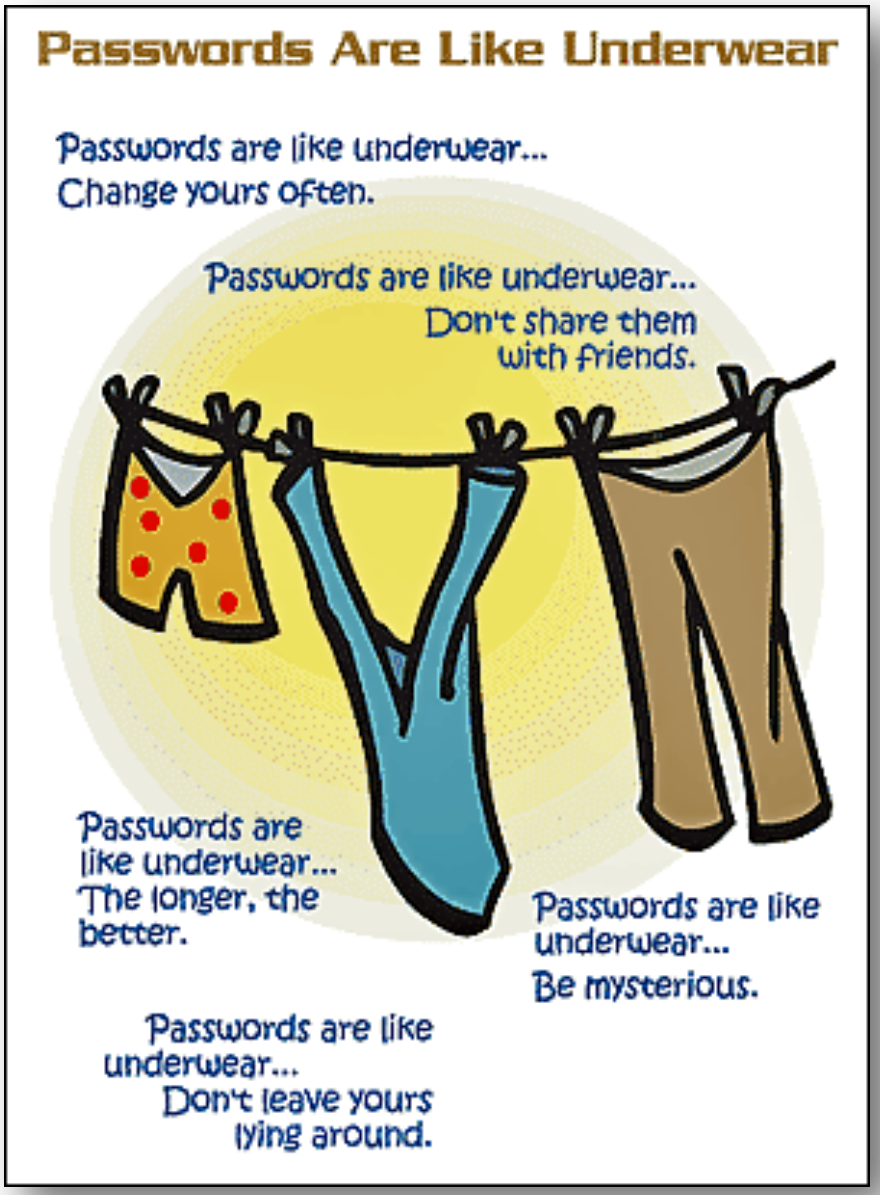
SQL Injection and Solutions

- SQL injection attacks are important security threat that can
  - Compromise sensitive user data
  - Alter or damage critical data
  - Give an attacker unwanted access to DB

- Key Idea: Use solutions consistently!
  - Whitelisting input validation & escaping
  - Prepared Statements with bind variables

- Limit Privileges (Defense-in-Depth)
- Encrypt Sensitive Data stored in Database
- Harden DB Server and Host O/S

Password Security



Dictionary Attacks

- Offline: attacker steals file and tries combos
- Online: try combos against live system

Salting

- Salting – include additional info in hash
- Add third field to file storing random # (salt)

- Dictionary attack against arbitrary user is harder
  - Before Salts: hash word & compare with password file
  - After Salts: hash combos of word & possible salts
- Ineffective against chosen-victim attack
  - Attacker wants to compromise particular account
  - Just hash dictionary words with victim's salt

Additional Password Security Techniques

- Several other techniques to help securely manage passwords: Mix and match ones that make sense for particular app
  - Strong Passwords
  - Honeypots
  - Filtering
  - Aging
  - Limiting Logins
  - Artificial Delays
  - Last Login
  - Image Authentication
  - One-Time Passwords

“ lemotdepassedeyoutube. ” -> “the password of youtube.”

Best Practice

- Hashing passwords: don’t store in clear
- Dictionary Attacks: try hashes of common words

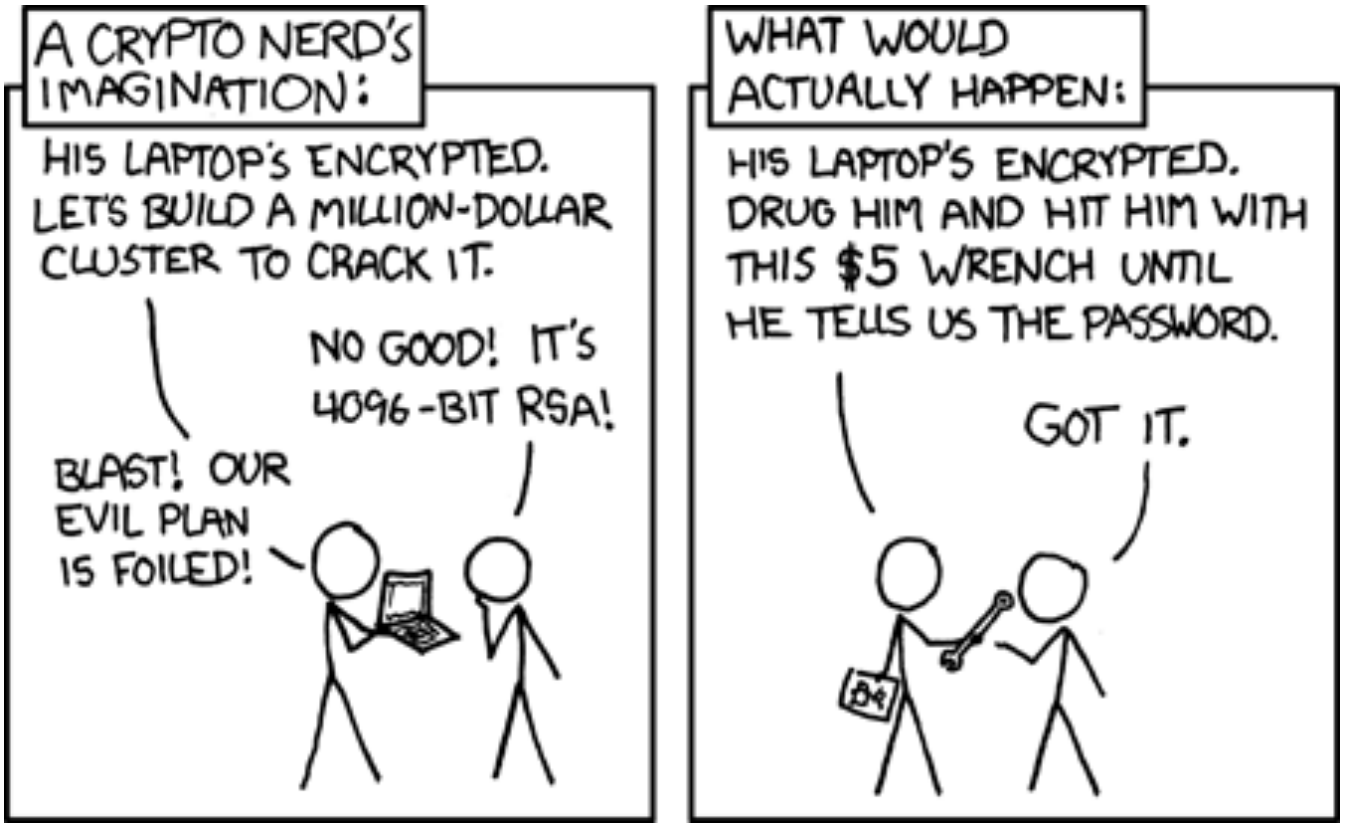
- Salting: add a random #, then hash
  - Dictionary attack harder against arbitrary user
  - But doesn’t help attack against particular victim

- Other Approaches:
  - Image Authentication
  - One-time Passwords

Human Attacks

- Piggybacking and shoulder surfing
- Dumpster diving
- Social engineering
  - Gain trust of insider
    - People generally want to help somebody who is requesting help
    - People generally want to avoid confrontation

Human Attacks



Social Media

- Personal information found on Facebook, Twitter etc.

https://www.youtube.com/watch?v=82OcxRHylGc

Cryptography

Cryptography

- Is:
  - A tremendous tool
  - The basis for many security mechanisms
- Is not:
  - The solution to all security problems
  - Reliable unless implemented and used properly
  - Something you should try to invent yourself

Ancient Cryptography

- Spartans used a ribbon wrapped around a specific gauge cylinder and then wrote on the ribbon
  - When unwrapped, the ribbon appeared to hold a strange string of letters
  - Message could be read only when someone wrapped the ribbon back around the same gauge cylinder
- Romans used shift cipher
  - One letter of the alphabet is shifted a set number of places in the alphabet for another letter

Modern Cryptography

- Symmetric Cryptography

- Asymmetric Cryptography
  - Digital signatures
  - Certificate Authorities

Symmetric Encryption

- Sender and the receiver use same key
  - requires key management
  - key must be exchanged by other means

- Algorithms: DES, AES etc.

Asymmetric Encryption

- Sender and receiver use different keys
  - algorithm involves difficult math problems
  - also known as public key cryptography

- Goal: ensure secure communication
- Each party has public/private key pair
- Encryption used to ensure
  - confidentiality of communication
  - identity of parties

- Introduces digital signatures

- Algorithms: RSA, El-Gamal, ECC

Hashing

- Apply hash function to plain-text
  - is a special mathematical function that performs
  - one-way encryption to produce cipher-text

- Common uses of hashing:
  - storing computer passwords
  - ensuring message integrity

- MD5, SHA-1, SHA-256 etc.

Trust Models and PKIs

- PKI: Framework to enable secure communication

- PKI components:
  - Keys: public and private
  - Certificates, to hold keys
  - Authorities, to register & verify

Trust Models and PKIs

- CA is trusted authority for certifying an individual's identity

- CA issues digital certificate
  - certifies association between subject's identity and a public key
  - private key that is paired with the public key
  - in the certificate is stored separately
  - certificate is signed with CA's private key

TLS/SSL & CAs

- Secure Sockets Layer (SSL)
- Transport Layer Security (TLS)
  - Most common means of interacting with a PKI and certificates.

- Ensures privacy between communicating applications and their users on the Internet.

Pretty Good Privacy (PGP)

- Alternative to current PKI model
  - Based on: Web of trust
  - Decentralized grass-roots model

- Created by Philip Zimmermann in 1991

- Implemented by
  - PGP
  - OpenPGP
  - GnuPG

ASSIGNMENT 2 – PGP

- Create your own key pair (public & private)
- Name your public key as
  - name_surname_stdid.pub
- Bring your public key to the course in a USB flash drive. (**DEADLINE: TBD**)
- Exchange your public key with me.
- Send an encrypted and signed e-mail to the instructor. (**DEADLINE: TBD**)

SSH Trust Model

- Trust on First Use (TOFU)

Key Complexity

- Key complexity = degree of security of the system
  - Key complexity = number of possible key values
  - Key space depends on size of key value
    - 48bit vs. 64bit vs. 128bit vs. 192bit vs. 256bit

- Brute-force attack: attempting every possible key

Remarks

National Geographic – Hack the System
https://www.youtube.com/watch?v=j_ZCa6PMRLQ

Corporate Security

Security

- Physical Security
- Technological Security
  - Application Security
  - Operating System Security
  - Network Security
- Policies & Procedures

- All Three Required

Physical Security

- Protecting against information leakage and document theft
- Limit access to physical space to prevent asset theft and unauthorized entry
- Ex: **Dumpster Diving** – gathering sensitive information by sifting through the company's garbage.

OS & Network Security

- Apps (e.g. servers) use OS for many functions
- OS code likely contains vulnerabilities
  - Regularly download patches to eliminate (e.g. Windows Update for critical patches)
- Network Security: mitigate malicious traffic
- Tools: Firewalls & Intrusion Detection Systems

Operating System

- Apply all service packs and patches
  - install anti-virus software
  - install anti-spyware software
  - install spam filter
  - configure automatic notification and update
- Create and enforce password policy
  - password aging
  - password audits
  - no password recycling

Operating System

- Restrict permissions on files and directories
  - remove all unnecessary file shares
  - possible remove File And Printer Sharing protocol
  - set appropriate ACLs on all necessary file shares
- Disable unnecessary services
- Remove unnecessary users
- Disable or remove unnecessary programs
- Enable security event auditing

Network Hardening

- Routers, switches, and access points
  - remove defaults: ID/user/password
  - control network access
  - apply updates and patches

- Traffic filtering: firewall
  - elements: rules that accept vs. deny traffic
  - bandwidth/throughput sensitive
  - advanced rules to control traffic:
    - port/service specific
    - stateful traffic analysis
    - intelligent decisions

Application Security

- No flaws in identity verification process
- Configure server correctly
  - local files
  - database content
- Interpret data robustly

Application Hardening

- Secure applications against local and Internet-based attacks
  - Remove unneeded functions or components
  - Restrict access where you can
  - Make sure application is kept up-to-date with patches

Policies & Procedures

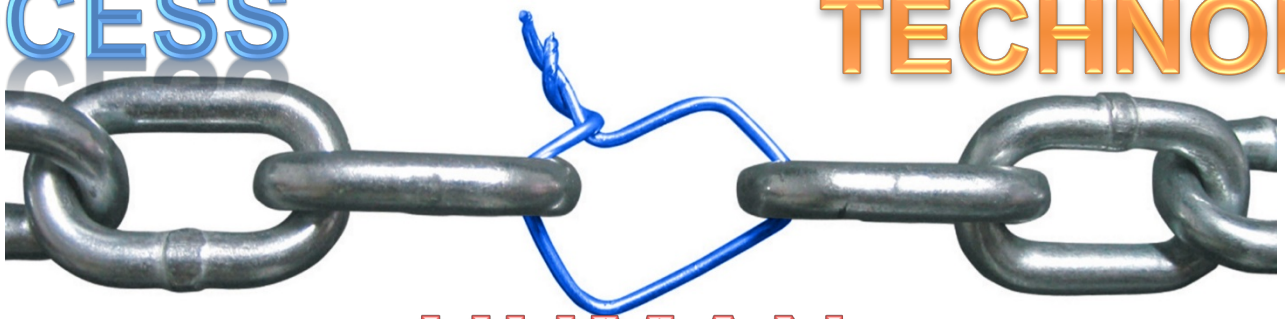
- Ex: Social engineering attack - taking advantage of unsuspecting employees (e.g. attacker gets employee to divulge his username & password)
- Guard sensitive corporate information
- Employees need to be aware, be educated to be somewhat paranoid and vigilant

Servers

- Web
- Mail
- FTP
- DNS
- File and Print
- Directory

PROCESS

TECHNOLOGY



HUMAN

Technology

- Firewall
  - Next Generation Firewall
- IDS/IPS
- Network Access Control (NAC)
- Data Leakeage Prevention (DLP)
- Endpoint security
- Antivirus, antispyware
- Content Filtering (URL vb.)
- DDoS prevention
- E-mail security gateway

EOF