TSIT02 Computer Security
Lecture 1: Introduction

Jonathan Jogenfors
Information Coding Group
Department of Electrical Engineering
Coursebook, examination

- 12 lectures
- A lab course
- Written exam

- Course book is used in some lectures (freely available online)
New for this year

• Earlier course evaluations: Decent, can be improved
• This time we plan to make the course better!
• Lectures: Removed some old boring stuff, added new exciting topics!
• Labs: Old labs were mostly irrelevant. Entirely new lab course!
• Big focus on making the course more relevant
• Guest lectures have been very popular!
The following form's parameters are concatenated to a string that will be passed to a SQL server. This means that the data can be interpreted as part of the code.

The objective here is to modify the result of the query with SQL Injection so that all of the table's rows are return. This means you want to change the boolean result of the query's WHERE clause to return true for every row in the table.

The easiest way to ensure the boolean result is always true is to inject a boolean OR operator followed by a true statement like 1 = 1.

If the parameter is interpreted as a string, you can escape the string with an apostrophe. That means that everything after the apostrophe will be interpreted as SQL code.

Hide Lesson Introduction

Exploit the SQL Injection flaw in the following example to retrieve all of the rows in the table. The lesson's solution key will be found in one of these rows. The results will be posted beneath the search form.

Please enter the user name of the user that you want to look up:

```sql
SELECT * FROM tb_users WHERE username = 'jim' OR 1=1;
```

Submit

**Search Results**

<table>
<thead>
<tr>
<th>User Id</th>
<th>User Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>12345</td>
<td>user</td>
<td>Try Adding some SQL Code</td>
</tr>
<tr>
<td>12346</td>
<td>OR 1 = 1</td>
<td>Your Close. You need to escape the string with an apostrophe so that your code is interpreted</td>
</tr>
<tr>
<td>12543</td>
<td>Fred Mteni</td>
<td>A lecturer in DIT Kevin Street</td>
</tr>
<tr>
<td>14232</td>
<td>Mark Danihan</td>
<td>This guy wrote this application</td>
</tr>
<tr>
<td>61523</td>
<td>Cloud</td>
<td>Has a Big Sword</td>
</tr>
<tr>
<td>82642</td>
<td>gwidshs@ab</td>
<td>Lesson Completed. The result key is 3c17f6bf34080979e0cbed5572e898c07c0ac3f04a7b7c17c7b13d6ce26b738</td>
</tr>
</tbody>
</table>

Start
Organisation

- Jonathan Jogenfors (Me, Main lecturer)
- Jan-Åke Larsson (Lectures, Examinationator)
- Niklas Johansson (Labs)
- Ingo Hölscher (Lectures 2 and 3: Authentication)
- Sebastian Wärmländer (Lecture 10: legal perspectives)
Communication

- Course homepage: http://www.icg.isy.liu.se/courses/tsit02/
- Lisam: https://goo.gl/f5AI0M
- E-mail: Check your LiU email regularly.

Lab sign up starts right after the lecture.
October 2016: IoT-botnet

21 Hacked Cameras, DVRs Powered Today’s Massive Internet Outage

A massive and sustained Internet attack that has caused outages and network congestion today for a large number of Web sites was launched with the help of hacked “Internet of Things” (IoT) devices, such as CCTV video cameras and digital video recorders, new data suggests.

Earlier today cyber criminals began training their attack cannons on Dyn, an Internet infrastructure company that provides critical technology services to some of the Internet’s top destinations. The attack began creating problems for Internet users reaching an array of sites, including Twitter, Amazon, Tumblr, Reddit, Spotify and Netflix.
Availability can also be security!
The 1960s — the dawn of computer security

- Multi-user systems emerge
- Users need to be restricted, so use authentication (Ch 4)
- Your data needs to be protected from other users
- “Protection rings” (Ch 4.3) is from this period
The 1970s — the era of mainframes

- “the Anderson report”, mainly to protect classified information
- the Bell-LaPadula formal model (Ch 8) regulates access to classified information
- Larger storage (35 MB) enabled data processing for US government departments
- Access control mechanisms (Ch 4) were created
The 1970s — the era of mainframes

- Encryption was also needed, hence the creation of the Data Encryption Standard (Ch 5)
- Proposal of Diffie-Hellman public key distribution (Ch 5)
- Database security was starting to matter (Ch 4)
- The legal system was adapted
The 1980s — the rise of personal computers

- Single user machine: multi-level multi-user security irrelevant
- “The Orange Book” is published
- Security research continued: database security for entering data
- The Clark-Wilson and Chinese Wall models were created for commercial systems, in the late eighties (Ch 9)
- The first worms and viruses appear
The 1980s — the rise of personal computers

- Single user machine: multi-level multi-user security irrelevant
- “The Orange Book” is published
- Security research continued: database security for entering data
- The Clark-Wilson and Chinese Wall models were created for commercial systems, in the late eighties (Ch 9)
- The first worms and viruses appear
The 1980s — the rise of personal computers

- The “Morris worm” of 1988 infected 5-10% of all machines connected to the internet
- Used a buffer overrun in the *fingerd* daemon of VAXes running BSD Unix

```
push1 $68732f            push '/sh, <NUL>'
push1 $6e69622f          push '/bin'
mov1 sp, r10            save stackp in r10 (string beginning)
push1 $0                 push 0 (arg 3 to execve)
push1 $0                 push 0 (arg 2 to execve)
push1 r10                push string beginning (arg 1 to execve)
push1 $3                 push argc
mov1 sp, ap              set argv to stackp
chmk $3b                 perform 'execve' kernel call
```
The 1990s — the era of the internet

- Creation of the Hypertext transfer protocol
- Security = communication security = strong cryptography (Ch 16)
- PC now connected, so needs software security, an example is the Java security model (Ch 4)
- Denial-of-service attacks appeared and needed to be protected against (Ch 21)
The 2000s — the era of the web

- Tech from the 90’s but much larger user space
- Commercial applications
- SQL injection, cross-site scripting (Ch 23), attacks on DNS (Ch 21)
- Attracts criminals
- More low profile attacks
- ...but higher economical losses
The 2010s — the era of the cloud/social networks

- Online storage/communication service
- Can only you access your data?
- Is it the same when you retrieve it?
- Can you always retrieve it?
- Many security issues in social networks
The 2010s — Internet of Things

- Smart phones that store our lives
- More and more connected devices – Internet of Things
- Our devices know everything about us . . . Privacy issues
- Cars can be hacked (Jeep)
- . . . and car manufacturers cheat with software (Dieselgate)
- Bitcoin
- Government surveillance – See: Snowden leaks
- Ransomware trojans are here to stay (Cryptolocker, Cryptowall)
Your files are encrypted.
To get the key to decrypt files you have to pay 750 USD/EUR. If payment is not made before, the cost of decrypting files will increase 2 times and will be 1500 USD/EUR.
Prior to increasing the amount left:
42h 48m 35s
Your system: Windows 7 (x64)  First connect IP: 203.182.204.16  Total encrypted files.

We are present a special software - CryptoWall Decrypter - which is allow to decrypt and return control to all your encrypted files.
How to buy CryptoWall decrypter?

1. You should register Bitcoin wallet (click here for more information with pictures)
2. Purchasing Bitcoins - Although it’s not yet easy to buy bitcoins, it’s getting simpler every day.

Here are our recommendations:
Computer security

- Computer security deals with the *prevention* and *detection* of *unauthorized* actions by users of a computer system.

- Computer security is concerned with the measures we can take to deal with *intentional* actions by parties behaving in some unwelcome fashion.
Other words similar to security

- In general engineering, defending against accidents and other malfunctions is called reliability.
- Robustness, resilience, safety, ...
- There are differences in what you stress using these words instead of security, but no strictly differentiating definitions.
- Safety is slightly different and means “no threat to human life or health.”
- Svenska: “Säkerhet” betyder både “Security” and “Safety”
Terminology: Prevention-Detection-Reaction

- Prevention aims to hinder damage to your assets
  - Typical tools are encryption, firewalls, ...
- Detection aims to discover damage to your assets, how it has been damaged, and who caused the damage
  - Typical tools are IDSs, digital signatures, ...
  - Can also be used to detect that an attack is on its way
- Reaction aims to mitigate damage, recover assets, and enhance existing protection
Example: Prevention-Detection-Reaction

Prevention aims to hinder damage to your assets

- Use encryption when placing an order using your credit card number
- Merchant should perform checks (delivery address for example)
- Don’t use card number on the internet
Example: Prevention-Detection-Reaction

Detection aims to discover damage to your assets, how it has been damaged, and who caused the damage

- A transaction you did not authorize appears on your card statement
- Amount and place of withdrawal (delivery address) can be used to track the perpetrator
Example: Prevention-Detection-Reaction

Reaction aims to mitigate damage, recover assets, and enhance existing protection

- Report incident, invalidate card
- Cost is covered by card holder, merchant, or card issuer depending on cause for vulnerability
- Stop using card on unencrypted connections, ...
Terminology: CIA

Three traditional areas of computer security

Confidentiality: Only authorised persons can read the protected information

Integrity: Only authorised persons can write or change the protected information

Availability: Authorized persons can read or write the information (in a timely manner)
CIA på svenska

Tre traditionella områden av datasäkerhet

Sekretess: Hålla data hemliga
Tillförlitlighet/Riktighet: Data ska vara oförvanskade
Tillgänglighet: Data ska finnas tillgängliga

OBS: Blanda inte ihop I-kravet med ett annat svenskt ord!
Terminology: Confidentiality

- We want to prevent that unauthorised persons read sensitive data
- Historically, confidentiality and security were closely linked, and are sometimes thought to be the same even now
- May refer to hiding information, but also to hiding that information exists
- Note that “access” is not a suitable verb, since it means both “read” and “write/change”.
Tools for confidentiality

- Physical access restrictions
- Computer theft precautions (alarms etc.)
- Access control in the computer systems
- Encryption in communication and storage
- Bug-free programs
Tools for confidentiality: Encryption

- A message is encrypted using a key
- The idea is that the message should be kept secret to those who do not have the key
- There are variants, where everyone can encrypt but only some can decrypt
Tools for confidentiality: Encryption

- A message is encrypted using a key
- The idea is that the message should be kept secret to those who do not have the key
- There are variants, where everyone can encrypt but only some can decrypt
Tools for confidentiality: steganography

- In this case, the message is hidden inside another message
- Usually there is no secret key in this case
- The purpose is to conceal the existence of the message
- A modern example is al-Qaeda hid secret documents in a video found in Berlin in 2011 (NSFW)
Tools for confidentiality: steganography

- In this case, the message is hidden inside another message
- Usually there is no secret key in this case
- The purpose is to conceal the existence of the message
- A modern example is al-Qaeda hid secret documents in a video found in Berlin in 2011 (NSFW)
Terminology: Integrity

- We want to prevent that unauthorised persons write or change sensitive data
- A good example of how difficult this can be is the “telephone game” (viskleken)
- Historically, integrity was ensured by writing in un-erasable ink and appending a signature (“blue-ink signature”), a seal, or a stamp
- More recently, integrity could be ensured by writing in unchangeable storage, WORM memory
Tools for integrity

- Physical access control
- Computer-based access control
- Checksums: values that can be used to check for random changes, like transmission or storage noise, “bitrot”
- Protecting against intentional changes requires stronger (cryptographic) tools
- Backups are used to restore data if unauthorized changes are noticed
Cryptographic tools for integrity

- A digital signature or MAC created using a key
- Integrity is checked by detecting changes in stored or transmitted data, by comparing a newly generated verification item with the old one
- The idea is that it should be hard to create the correct verification item without the key
Confidentiality and integrity

Confidentiality and integrity protection are often based on the same techniques.

- No physical access to storage media
- Access control on all logical access paths to data
- Cryptographic techniques
Terminology: Availability

- ISO7498-2: “The property of being accessible and usable upon demand by an authorized entity”
- Basically, to prevent denial-of-service
- Not about data being unavailable to unauthorized users, which is really C+I
Tools for Availability

- Protection against physical threats and attacks
  - Uninterruptable Power Supplies
  - Fire precautions
  - Flooding precautions
  - Temperature and humidity control
  - Storm resistant buildings
- Hindering secondary damage from data integrity attacks
  - Backups, backups and backups
Tools for Availability

- Countermeasures against system overload
  - Redundancy in server farms, or disks
  - Equipment that can detect and counter DOS attacks
- Countermeasures against system crashing
  - Check all user input for “out of bounds” values
  - Switch off unused services and functions
  - And of course follow and install supplier updates and patches
Terminology: CIA

Three traditional areas of computer security

Confidentiality: Only authorised persons can read the protected information

Integrity: Only authorised persons can write or change the protected information

Availability: Authorized persons can read or write the information (in a timely manner)
Assurance, Authenticity, and Anonymity (AAA)

- These are related to CIA; their internal relation is different
- Assurance: can we trust systems/people to behave as expected?
- Authenticity: is an issued statement/permission/policy/... genuine?
- Anonymity: can records/transactions not be tied to a particular individual?
Assurance

- In computer security this refers to management of trust
- Trust is difficult to quantify
- But trust, and management of trust, is essential
  - Can we trust that the OS is bug-free?
  - Can the computer trust that the user is who he claims to be?
  - Can the movie site trust you won't redistribute the movie?
  - Can the computer trust that the IP received from a name lookup is correct?
“Tools” for assurance

- In computer security this refers to management of trust
- Trust is difficult to quantify
- But trust, and management of trust, is essential

To ensure trust, we first specify

   Policies: How are people/systems expected to behave?
   Permissions: What actions are people/systems allowed to perform?

Then we put in place

   Protections: are mechanisms that enforce permissions
Authenticity

- Authenticity refers to the ability to determine that statements, policies and permissions are authentic
- Different from authentication
- Not quite integrity either
- Integrity refers to integrity of data in the system
Terminology: non-repudiation

- Repudiation means denial
- So non-repudiation is the inability to deny issuing a statement
- The most common example is an order of goods, but there are many other examples in computer security
- Inside computer security, non-repudiation is how to provide unforgeable evidence of an action (or the validity of a contract)
Non-repudiation, and accountability

- If the worst occurs, we want to be able to trace what has happened and who did it
- We want to hold the perpetrator accountable

- As prevention, it is aimed to dissuade attackers
- As compensation, it is aimed to recover losses from the attack
- As improvement, it is aimed to tell us how to harden our system
Tools for accountability: Audit trail, and digital signatures

- An audit trail is primarily to enable accountability
- A secondary goal is to find the vulnerability that enabled an attack
- Another goal is to find and/or restore changes
- Digital signatures is also a tool, binding an entity to an issued statement, or an action
The third A: Anonymity

Our identity is tied to the online transactions we perform:

- medical records
- purchases
- legal records
- email
- browsing history
Tools for anonymity

- On the web: proxies. Proxies are trusted agents that act for an individual so that the actions cannot be traced back to that individual. An example is The Onion Router

- Pseudonyms are fictional identities that fill in for real ones, and can only be traced back by a trusted entity

- Aggregation is combining data from many individuals so that statistical outputs cannot be traced back to individuals

- “Mixing” is a term that uses a quasi-random way of blending data so that searches can be performed without revealing individual identities
Security policy

- A statement that defines the security objectives of an organization
  - it has to state what needs to be protected
  - it may also indicate how this is done

- To formulate such a policy you need to know
  - what needs to be protected
  - how it might be vulnerable
  - what threatens the vulnerability
  - and how the threats can be countered
Security policy, example on physical access

- Who has access to company premises?
- Are there restricted areas?
- Is access by key, card, security guard checkpoint, ...?
- Do you need to wear an ID badge (visible)?
- Must visitors be accompanied?
- Are their bags checked upon entry/exit?
- When are buildings locked?
- Who has access to keys?
Security policy, example on passwords

• How long should a password be?
• Are all-lowercase ASCII passwords OK?
• Is there a dictionary attack on password creation, or on a regular basis in the system?
• How often is there a forced renewal?
• From where can users log in?
• What clients can be used?
Security policy

- A statement that defines the security objectives of an organization
  - it has to state what needs to be protected
  - it may also indicate how this is done

- To formulate such a policy you need to know
  - what needs to be protected
  - how it might be vulnerable
  - what threatens the vulnerability
  - and how the threats can be countered
Terminology: Assets

- First step: Identify assets, and their value
  - Hardware
  - Software
  - Data and information
  - Reputation
- The value is sometimes hard to estimate
- Monetary value, or in terms of (company) survival if asset is compromised
Terminology: Vulnerabilities

- Vulnerabilities are weaknesses that can be unintentionally or intentionally exploited to damage assets
- *Vulnerability scanners* are automated tools for detection (e.g., Nexpose)
- Risk analysis needs to measure their severity
- Examples:
  - Admin accounts with default passwords
  - Programs with unnecessary privileges
  - Programs with known flaws
  - Weak access control settings
  - Weak firewall configurations
Terminology: Threats

- A threat is a possible negative effect on your assets
- These can be categorized by the impact, e.g., Microsoft’s STRIDE threat model:
  - Spoofing identities
  - Tampering with data
  - Repudiation
  - Information disclosure
  - Denial of service
  - Elevation of privilege
- An alternative is to identify threats by source
Unintentional versus intentional threats

- Threats can be
  - unintentional, or
  - intentional (willfully caused by a human)
- Protection against accidents is usually covered by full protection against intentional events
- Full protection against accidents almost never covers all possibilities for intentional attacks
- This course is about protection from intentional attacks
Terminology: Attacks

- An *attack* is a sequence of steps needed to realize a threat.
- There may be several attacks that realizes a given threat.
- Often, an *attack tree* is constructed.

```
Get password
  ├── Guess password
  │    └── Guess online
  │         └── Get encrypted password
  └── Ask operator
      └── Guess offline
          └── Dictionary attack
  └── Spy password
      ├── In person
      │    └── Camera
      └── Microphone
```
Attackers’ goals and tools

- Financial gain, often C-attacks
  - Skimming, phishing, industrial espionage
- To show off, be noticed, often I-attacks
  - Virus writing, website defacement etc.
- Sabotaging opponent, often A-attacks
  - DDoS, system crashing etc.
Basic security problem

- The computer we use as a model in computer science is just an abstract construction.
- The real computer is a series of software layers interacting with each other and the hardware.
- Protection must deal with the real computer and the abstract model.
Basic conclusion

- For each possible attacker, find out the lowest level someone with his/her knowledge and resources can reach, subvert and control!

- Security measures above that layer are irrelevant for protection against that attacker. (Which does not mean that you should disregard known security holes in higher levels)
But always remember:

- If there is no possible attacker, there is no need to do anything!
- If an attacker can’t do anything you regard as damage (neither C, I nor A), there is no need to do anything!
- If security countermeasures cost more than the expected damage, make sure you know why you still want the protection! (Reputation? Laws? Uncertain about maximum damage?)
Security management

• Security is a people problem
• If it is awkward to follow security policy, it will likely not be followed
• This is especially true if the policy is not supported by the management
• A security policy should contain
  • Why security is important for individuals and the organization,
  • what is expected from individuals, and
  • which good practices they should follow.
• To anchor this in an organization, often a security awareness program is used
Security management

• A security policy should contain
  • Why security is important for individuals and the organization,
  • what is expected from individuals, and
  • which good practices they should follow.

• To anchor this in an organization, often a security awareness program is used
  • Example: NIKE employees have the sentence “Security is everyone’s responsibility” on the login screen
Measuring security

- Assign a numeric value to security
  - Severity of damage
  - Probability of damage

- Value is sometimes hard to measure
  - Number of open ports
  - Number of users (with weak passwords)
  - Number of unpatched programs
  - Cost of restoring data (in € or man hours)
  - Lost reputation (arbitrary units)
Measuring security

- Security costs do not generate revenue
  - cost must be motivated
  - ideally, a quantitative measure is needed
  - but often a qualitative measure is obtained
Standards for measuring security

- Some organizations have prescriptions for security management standards
  - Financial sector
  - Government departments
  - ... often regulated in law

- There are also codes of best practice, one example is ISO27002
  - Many topics including Security policy, HR security, Physical security, Access control, Incident response, Business continuity, Compliance, ...
  - Achieving compliance is a large task, and is not covered in its entirety in this course
Risk analysis, basic entities

Three entities must exist for a risk to exist:

- **Threat** — the cause of damage to asset(s)
- **Vulnerability** — the unwanted system property that enables the threat
- **Damage** — the adverse effect of an unwanted event
Example, basic entities

Theft by a pickpocket is a risk to you when:

- **Threat** — there can be pickpockets in places where you are

- **Vulnerability** — you carry your wallet where a skilled pickpocket can pick it in a crowd

- **Damage** — possible loss of your wallet with its contents
Properties of threats

- A threat is caused by a threat agent
- Accidental agents have no specific goal
- Deliberate threat agents have individual varying goals
- Each threat agent has individual resources, like time, computer power, knowledge etc.
- Threat agents have different probabilities of attacking your system
- Thus: Know the threats against your system!
Examples of threat agents

- Thunderstorms can be a threat to availability
  - Lightning strikes blindly
  - The power to destroy equipment is immense
  - The probability of a hit depends on geographical location, surroundings, any lightning rods etc. etc.

- An embezzling employee is a threat to finances via violations of data integrity
  - The employer provides computer resources
  - The employee has some system knowledge and can learn more
Identifying vulnerabilities

- Vulnerabilities are properties of your system
- A vulnerability can exist inside the system or in its close surroundings
- If you remove a vulnerability, its specific threat can no longer cause its damage (unless a parallel vulnerability remains)

Diagram:

- Hacker taking over private computers
- Patch against security hole not installed
- Your address blacklisted due to abusive spam

Threat  
Vulnerability  
Damage
Properties of vulnerabilities

- Vulnerabilities often form a chain of steps from the threat action to the damage.
- The links in the chain can be parallel: any of them will enable the final damage, or serial: all are necessary in turn for the final damage.
- Removing one serial link is sufficient, but with parallel links all must be removed.

<table>
<thead>
<tr>
<th>Hacker taking over private computers</th>
<th>Unused service running on computer</th>
<th>Patch against security hole not installed</th>
<th>Your address blacklisted due to abusive spam</th>
</tr>
</thead>
<tbody>
<tr>
<td>Threat</td>
<td>Vulnerability</td>
<td>Damage</td>
<td></td>
</tr>
</tbody>
</table>
Properties of vulnerabilities

- Removing one serial link is sufficient, but with parallel links all must be removed

Threat: Hacker taking over private computers
Vulnerability: Remote login possible without authentication
Damage: Patch against security hole not installed
        Your address blacklisted due to abusive spam
Identifying damages

- A damage is a loss of value to the system users
- Thus the damage is normally the effect outside the computer system when a threat occurs
- Sometimes you must analyse damages to the computer system itself, like corrupted access control data
- Primary damages fall into the CIA categories, which apply to data, but risk analysis treats the final effects of damaged data on system users
Properties of damages

• Normally you regard the damage to data as primary damage, and the effect of this as secondary damage

• It is secondary damage that counts (and costs)

• Some damages can be measurable in economic terms, but others concern ethics, reputation, laws etc.
Prevention-Detection-Recovery needs to be balanced

- Damage will eventually be detected (maybe indirectly, but still)
- If recovery at that time is cheaper/better than prevention would have been, don’t do prevention
- Early detection may save you expensive prevention
- What is “cheap” should be evaluated over the whole system life
- Remember to follow laws, and to include costs for possible loss of reputation
Strict risk analysis

• In risk analysis we evaluate the possible damage and magnitude of the threat in order to find out if the cost of removing a vulnerability is warranted

• If we gain more by introducing the countermeasure than the countermeasure costs us, we should use the countermeasure
Calculating the risk cost

- Calculate the average risk cost per year, \( r \)
- First find the average cost of the damage every time the threat causes a damage, \( d \)
- Then find the average number of times per year that you can expect the threat to cause a damage, \( f \)
- The risk cost \( r = f \cdot d \)
Using the risk cost

- Calculate the average risk cost per year before introducing a countermeasure, $r_b$
- Calculate the average risk cost per year after introducing a countermeasure, $r_a$
- Calculate the average cost per year for the countermeasure, $c$
- Introduce the countermeasure if $c < r_b - r_a$
Qualitative risk analysis

- Estimate roughly the magnitude of damage:
  - Negligible?
  - Bearable?
  - Serious?
  - Catastrophic?

- Estimate roughly the probability:
  - Almost impossible?
  - Possible?
  - Likely?
  - Almost certain to happen?
Qualitative risk analysis

- Put your estimates in a grid:

<table>
<thead>
<tr>
<th></th>
<th>Almost impossible</th>
<th>Possible</th>
<th>Likely</th>
<th>Almost certain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catastrophic</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Serious</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bearable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Negligible</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Qualitative risk analysis

- Treat events in order of priority!

<table>
<thead>
<tr>
<th>Catastrophic</th>
<th>Almost impossible</th>
<th>Possible</th>
<th>Likely</th>
<th>Almost certain</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serious</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bearable</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Negligible</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Summary of first lecture

- Models for security
- CIA
- Security policy
- AAA
- Threat, Vulnerability, Damage
- Measuring security
- Risk analysis
Next two lectures: Authentication

- Authentication, to certify that an entity is who/what it claims to be

- Authorization is a different concept, and is to allow or deny a request from an entity based on what permissions that entity has

- Biometry: A method for authentication