Identifying Threats & Threat Risk Modeling

고려대학교 (Korea Univ.)

사이버국방학과 · 정보보호대학원 (CIST)

보안성분석평가연구실 (Security Analysis aNd Evaluation Lab.)

김 승 주 (Seungjoo Kim)

www.kimlab.net

보안성분석평가연구실

Security Analysis aNd Evaluation Lab

www.KimLab.net / www.SecEng.net

김승주 교수 (skim71@korea.ac.kr)

로봇융합관 306호

연구분야

- Security Eng. for High-Assurance Trustworthy Systems
- High-Assurance Cryptography
- Security Testing (including End-to-End Provable Security, Formal Verification) and Security Evaluation (e.g. CMVP, CC, C&A, SSE-CMM)
- Usable Security

주요 경력 :

1990.3~1999.2) 성균관대학교 공학 학사·석사·박사
1998.12~2004.2) KISA 암호기술팀장 및 CC평가1팀장
2004.3~2011.2) 성균관대학교 정보통신공학부 부교수
2011.3~현재) 고려대학교 사이버국방학과·정보보호대학원 정교수
Founder of (사)HARU & SECUINSIDE

前) 육군사관학교 초빙교수
前) 선관위 DDoS 특별검사팀 자문위원
前) SBS 드라마 '유령' 및 영화 '베를린' 자문 / KBS '명견만리' 강연
現) 한국정보보호학회 이사
現) 대검찰청 디지털수사 자문위원
現) 개인정보분쟁조정위원회 위원

- '96: Convertible group signatures (AsiaCrypt)
- '97: Proxy signatures, revisited (ICICS): 670회 이상 인용
- '06: 국가정보원 암호학술논문공모전 우수상
- '07: 국가정보원장 국가사이버안전업무 유공자 표창
- '12, '16: 고려대학교 석탑강의상
- '13: Smart TV Security (Black Hat USA): 스마트TV 해킹(도청·도촬) 및 해적방송 송출 시연

주요 R&D 성과

삼성전자와 공동으로
국내 최초 프린터복합기 보안 인증 획득 (2008년)

LG전자와 공동으로
국내 최초 스마트TV 보안 인증 획득 (2015년)

Security Engineering

- Security Engineering is the art and science of discovering users' information protection (   ) and then (   ) and (   ) information systems, (   ), so they can safely resist the forces to which they may be subjected

(National Security Agency, 2002)

"Threat" Risk Modeling

- SE is unlike other engineering fields in the respect that the majority of the "forces" to be modelled are caused by human threat actors with deliberate intent, as opposed to forces due to natural and accidental causes. Thus, the first major hurdle facing security engineering is to define and maintain a threat model that can be used to
( ).

Threat "Risk" Modeling

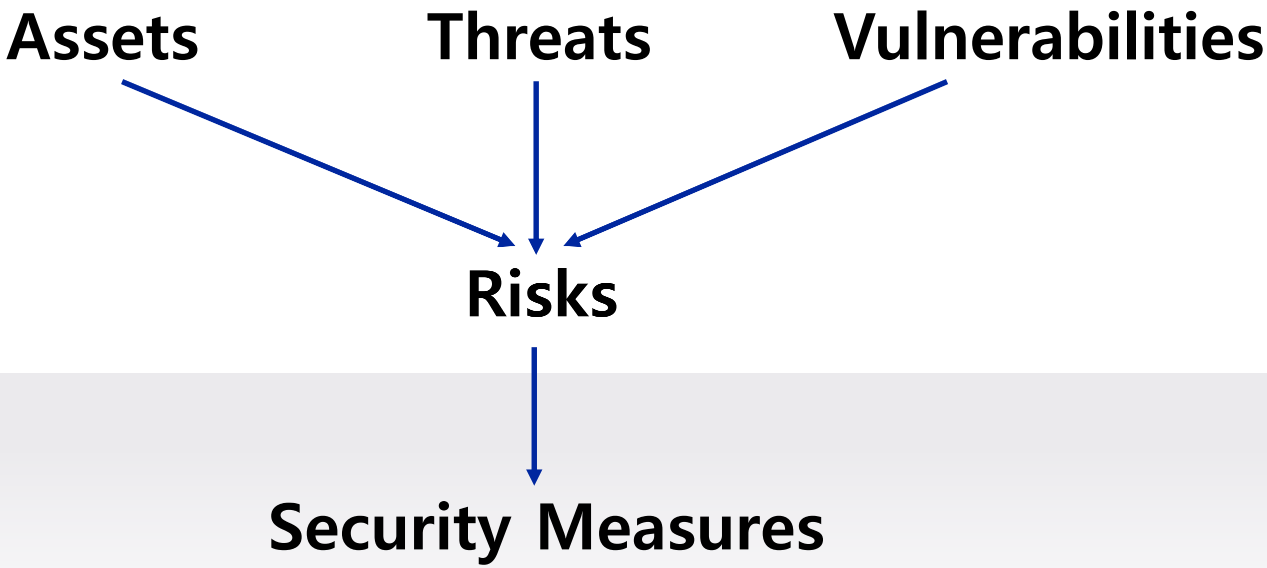
- It is a security analysis to determine the (   ) security risks to a system. The goal is to (   ) the risk to an acceptable level by determining threats to mitigate and the steps to mitigate the identified threats.

Threat Risk Modeling

- **Commercial**
    -
    -
    -
- **Standards**
    -
    -
    -

KOREA UNIVERSITY

Threat Risk Analysis & Management



Threat Risk Analysis & Management

- (   ) involves the identification and assessment of the levels of risk, calculated from the
  - Values of assets
  - Threats to the assets
  - Their vulnerabilities and likelihood of exploitation
- (   ) involves the identification, selection and adoption of security measures justified by
  - The identified risks to assets
  - The reduction of these risks to acceptable levels

KOREA UNIVERSITY

Goal of Threat Risk Analysis

- All assets have been identified
- All threats have been identified
  - Their impact on assets has been valued
- All vulnerabilities have been identified and assessed

Problems of Measuring Risk

Businesses normally wish to measure in money, but

- Many of the entities **do not** allow this
  - Valuation of assets
    - Value of data and in-house software - no market value
    - Value of goodwill and customer confidence
- Likelihood of threats
  - How relevant is past data to the calculation of future probabilities?
    - The nature of future attacks is unpredictable
    - The actions of future attackers are unpredictable
  - Measurement of benefit from security measures
    - Problems with the difference of two approximate quantities
    - How does an extra security measure affect a ~10^{-5} probability of attack?

KOREA UNIVERSITY

Risk Levels

- Precise (   ) values give a (   ) precision
- Better to use (   ), e.g.
  - High, Medium, Low
    - High: major impact on the organisation
    - Medium: noticeable impact ("material" in auditing terms)
    - Low: can be absorbed without difficulty
  - 1 - 10
- Express (   ) values in (   ), e.g.
  - For a large University Department a possibility is
    - High
    - Medium
    - Low

Threat Risk Analysis Steps

1. Decide on Scope of Analysis and Set the System Boundary
2. Identification of Assets
3. Identify Threats and Possible Vulnerabilities by using STRIDE
   - ✓ Develop Attack Scenarios by Threat Trees or Attack Trees, etc.
4. Rank Threat Risks based on Probability and Impacts (Threat Probability or Risk Assessment)
5. Responses to Risk (Risk Mitigation or Countermeasures)

Defining the System Boundaries

Defining the System Boundaries

- Defining system boundaries **to be protected** and information security/IA **responsibilities**.
  - Crucial step! The system definition should be reviewed and approved by (   )

- It is comprised of :
  -
  -
  -

KOREA UNIVERSITY

Types of Diagrams

- **Goal :** The goal of all the diagrams is to communicate how the system works, so that (   ) involved in threat modeling has the (   ) understanding.
    - Lead to a substantial improvement in the security of those components.
- **Types :**
    -
    -
    -
    -

KOREA UNIVERSITY

Identification of Assets

Identification of Assets

- Types of asset
  - Hardware
  - Software : purchased or developed programs
  - Data
  - People : who run the system
  - Documentation : manuals, administrative procedures, etc
  - Supplies : paper forms, magnetic media, printer liquid, etc
  - Money
  - **Intangibles**
    - Goodwill
    - Organisation confidence
    - Organisation image

Determine Threats

IA Threats

Determine Threats

Step Objective : To identify threats for each data flow diagram element in the threat model.

- **Experts :** Brainstorming and other informal methods
- **Experts and Non-Experts :** STRIDE threat types
  - Based on Common Vulnerability and Exposures (CVE) (see http://cve.mitre.org for more information), etc.

KOREA UNIVERSITY

Determine Threats

1. **Threat lists**
   - Start with laundry list of possible threats
   - Work top down, and as you do, at each level of the diagram(s), work across something : ( ), ( ), ( )
2. **Grouping (e.g., STRIDE)**
   - Categorized list of threat types
   - Identify threats by type/category
3. **Optionally draw threat trees or attack trees**
   - Root nodes represent attacker's goals
   - Trees help identify threat conditions

Determine Threats

"InfoSec resources can best be applied only if guided by a **structured threat assessment process**."

A.Rathmell, "Assessing the IW threat from sub-state groups", Cyberwar 2.0: Myths, Mysteries and Reality, AFCEA International Press, 1998, 295–312.

MS's STRIDE Threat Model

- S
- T
- R
- I
- D
- E

MS's STRIDE Threat Model

- S
  - An adversary impersonates a different person and pretends to be a legitimate user to the system.
  - Spoofing attack is mitigated through authentication.

MS's STRIDE Threat Model

- T
  - Any data to the application or from the application should be secured so that it cannot be altered.
  - The application should validate all data received from the user before storing or using it for any processing.
  - An attacker should not be able to change data delivered to a user.
  - Data in the disk and any other storage media need to be protected.

MS's STRIDE Threat Model

- **R**
  - A dishonest user may dispute a genuine transaction if there is insufficient auditing or record keeping of their activity.
  - For example, a bank customer may say, "The signature on the check is forged and the money should be credited in my account!"
  - Applications need to have audit trails and systems by which the activity of a user can be proved beyond doubt.

MS's STRIDE Threat Model

- I
  - If it is possible for an attacker to publicly reveal user data, whether anonymously or as an authorized user, there will be an immediate loss of confidence and reputation.
  - Disclosure of proprietary or secured information may lead to serious financial loss.

MS's STRIDE Threat Model

- **D**
  - Application designers should be aware that their applications may be subject to a DoS attack.

MS's STRIDE Threat Model

- **E**
  - If an application provides distinct user and administrative roles, it is vital to ensure that the user cannot elevate his role to a higher privilege one.
  - All actions should be gated through an authorization matrix, to ensure that only the permitted roles can access privileged functionality.
  - The privileged access must be for the **minimum** duration it is necessary.

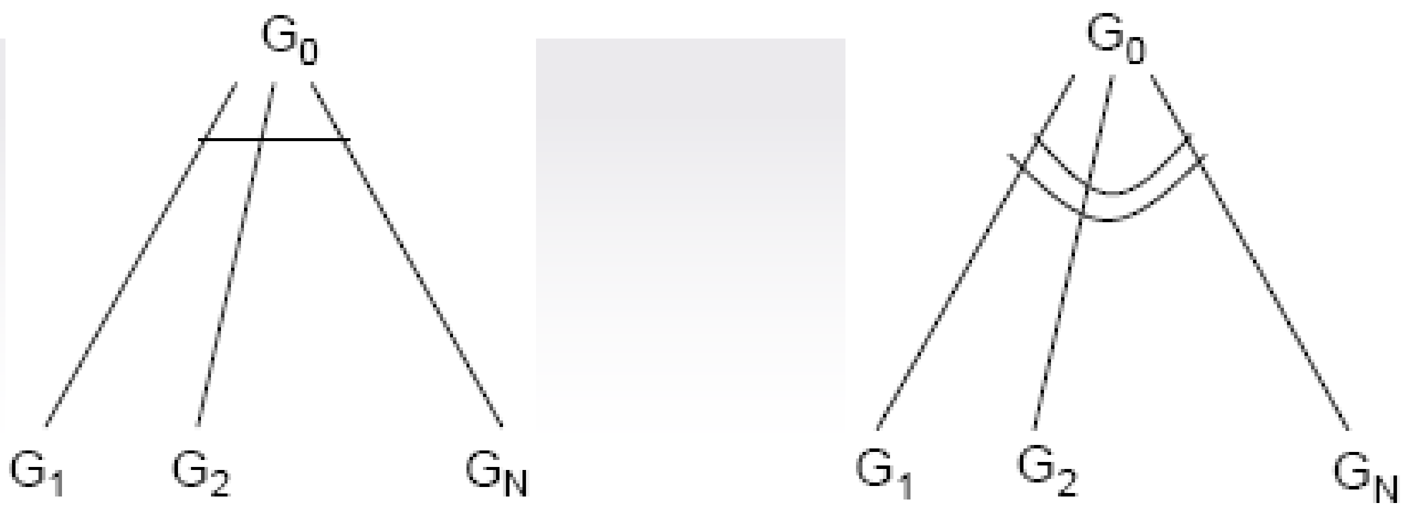
MS's STRIDE Threat Model

- **STRIDE-per-Element**

- **STRIDE-per-Interaction**

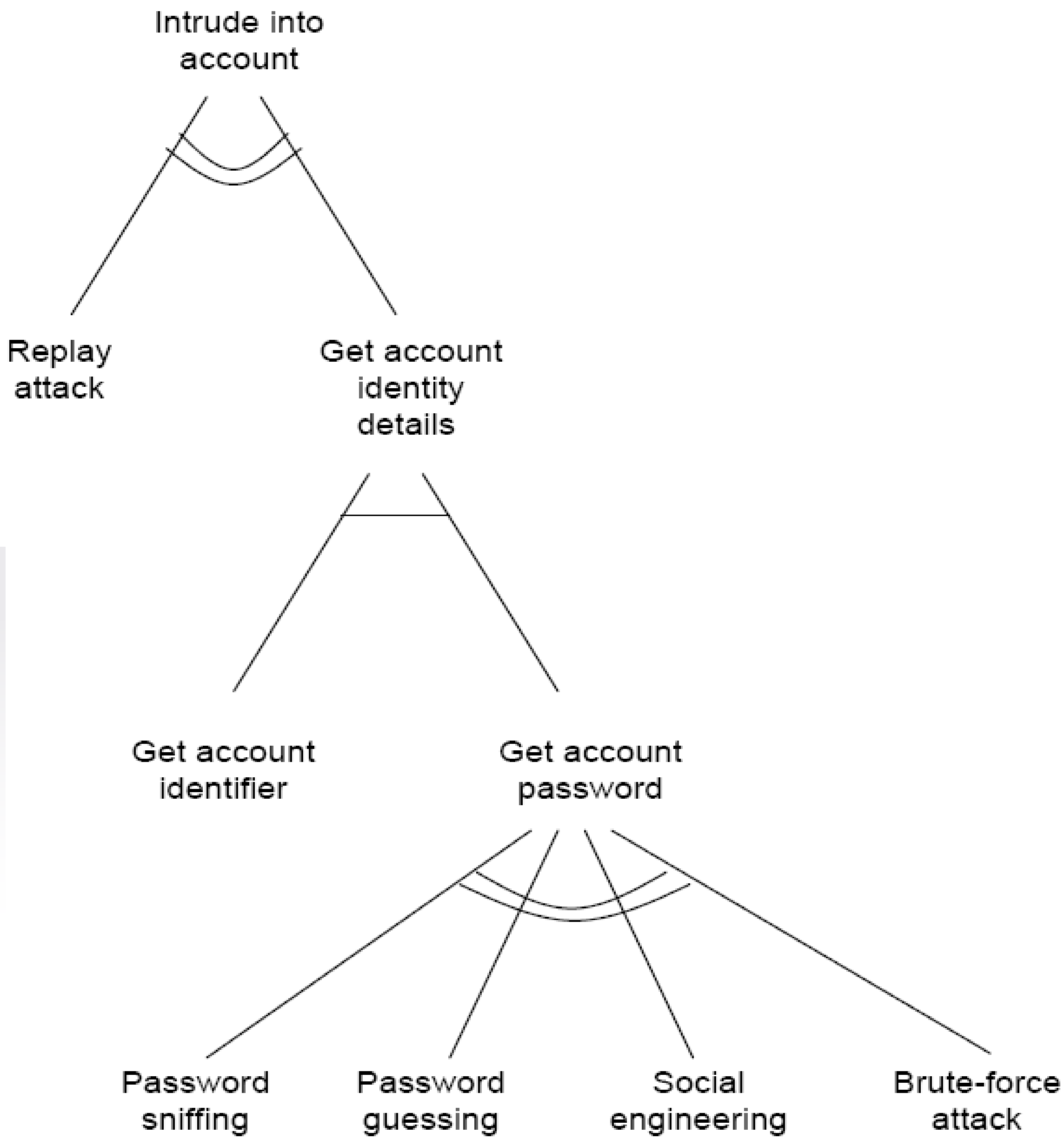
Attack Trees

- Attack tree is a tool to evaluate the system security based on various threats.
- Various vulnerabilities and compromises are used to build the attack tree.



(a) AND-decomposition and (b) OR-decomposition.

Attack Trees



Attack Trees

- **Spoofing identity**
  - Intrusion scenario can also be expressed as
    - (Replay Attack)
    - (Account-identifier, sniff-Password)
    - (Account-identifier, guessed-Password)
    - (Account-identifier, social-engineered-Password)
    - (Account-identifier, cracked-Password-through-brute-force)

Documenting Threats

- Document threats using a template

Intrusion into Account by Replay Attack	
Threat target	
Risk	
Attack techniques	
Countermeasures	

Intrusion into Account by Sniff Password	
Threat target	
Risk	
Attack techniques	
Countermeasures	

Rank Threats

Risk Assessment

- **(   ) Risk Analysis**
  - + probability theory based on mathematical theory
  - - quality of results depends on quality of inputs
  - - not always feasible
- **(   ) Risk Analysis**
  - + more applicable
  - - scaling based on judgements of security expert
  - e.g.) **DREAD** Risk Analysis Model

Simple Risk Analysis Model

Risk = Probability * Damage Potential

1-10 Scale

1 = Least probable
10 = Most probable

1-10 Scale

1 = Least damage
10 = Most damage

MS's DREAD Risk Analysis Model

- Greater **granulation** of threat potential
- Rates (prioritizes) each threat on scale of 1-15
- Developed and widely used by Microsoft

KOREA UNIVERSITY

MS's DREAD Risk Analysis Model

- D
- R
- E
- A
- D

- Risk =

KOREA UNIVERSITY

MS's DREAD Risk Analysis Model

	High (3)	Medium (2)	Low (1)
Damage potential			
Reproducibility			
Exploitabilty			
Affected users			
Discoverabilty			

MS's DREAD Risk Analysis Model

Threat	D	R	E	A	D	Sum
Intrusion Account (Replay Attack)						
Intrusion Account (Sniff Password)						

Responses to Risk

Risk Mitigation

Responses to risk

- (   ) it completely by withdrawing from an activity
- (   ) it and do nothing
- (   ) it with security measures

Security Measures

Possible security measures

- Transfer the risk, e.g. insurance
- Reduce vulnerability
  - Reduce likelihood of attempt
    - e.g. publicize security measures in order to deter attackers
    - e.g. competitive approach - the lion-hunter's approach to security
  - Reduce likelihood of success by preventive measures
    - e.g. access control, encryption, firewall
- Reduce impact, e.g. use fire extinguisher / firewall
- Recovery measures, e.g. restoration from backup

KOREA UNIVERSITY

Risk Management

- Identify possible security measures

- Decide which to choose
  - Ensure complete coverage with confidence that :
    - The selected security measures address (   ) threats
    - The results are (   )
    - The (   ) are commensurate with the risks

Iterate

- Adding security measures changes the system
  - (   ) vulnerabilities may have been introduced
- After deciding on security measures, (   ) the risk analysis and management processes
  - e.g. introduction of encryption of stored files may remove the threat to Confidentiality but introduce a threat to Availability
    - What happens if the secret key is lost?

KOREA UNIVERSITY

Case Study

Threat Modeling and Data Sensitivity Classification for Information Security Risk Analysis

Secure Electronic Elections – Case Study

Conference on Data Protection

December 2003

Belgrade, Serbia and Montenegro

Goran Obradović

Director of Technology

Chief Information Security Officer

goran@dvscorp.com

DOMINION VOTING SYSTEMS

Agenda

- ❖ *Problem Statement*
- ❖ *Anti Patterns in Info Security Practice*
- ❖ *Info Security Risk Analysis – The Journey*
- ❖ *Threat Modeling with examples in Electronic Voting Systems*
- ❖ *Current state-of-the-art electronic election systems*
- ❖ *Conclusions*
- ❖ *Q & A*

Problem Statement

It is not acceptable that only technical part of the team defines security requirements. Business stakeholder must be involved.

Events = Threats

Causes = Vulnerabilities

ROSI = Return on Security Investment

- *To secure an application or a system without spending excessive time and effort we are tempted to blindly apply security controls that have already been extensively used in practice*
- *However, without understanding security requirements common security controls can not provide adequate protection within the specific context*
- *We have to understand:*
  - ➢ *the real value of information resources that we need to protect*
  - ➢ *if an attacker has an interest to compromise our system*
  - ➢ *what are the events and causes that will have an unwelcome consequence upon our system*
  - ➢ *what will be risk mitigation techniques that will maximize our ROSI index and minimize overall threat probability or risk to an acceptable level*

DOMINION VOTING SYSTEMS

Info Security Anti Patterns

Some Common Security Patterns:

- Use firewalls

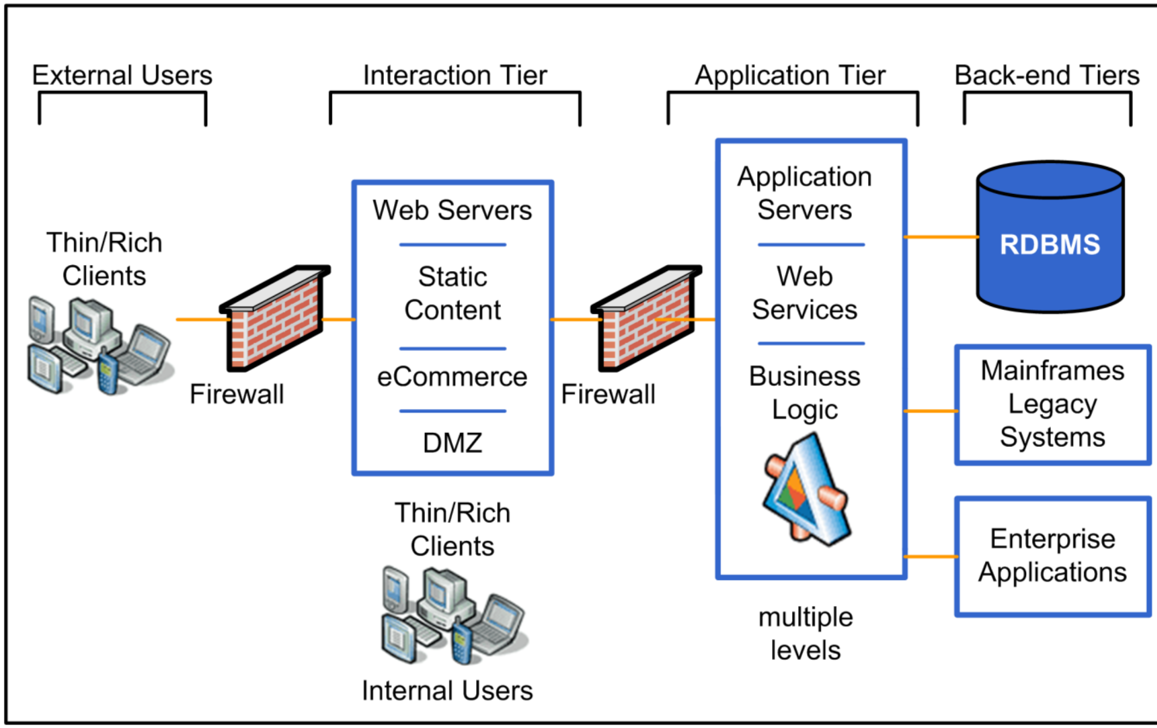
- Use SSL/TSL to encrypt everything

- Use X.509 Certificate authentication

- Customer does not know what security he needs

- We will use the latest version of the security product XYZ

N-Tier Enterprise Information System - Example



- Users tier – external (known/unknown) and internal users
- Interaction tier – Web Servers and presentation logic
- Application tier – Application Servers, Web Services and Business Logic
- Back-end tiers – DBMS, Legacy Mainframes, EA applications

Info Security Anti Patterns

N-Tier Enterprise Information System
Dental Patient Record

Original Patient Record

C:\Documents and Settings\gobrado\My Documents\NewFile.xml *

```xml

<?xml version="1.0" encoding="utf-8" ?>
<Patient name="John Johnson" id="123-456-789">
    <Address>...</Address>
    <Visit date="11/11/2002" id="ElevnthNovemberRegular">
        <DoctorCheckUp>
            <Observation DoctorName="Michael Carter" id="ElNovObserv">
                <PatientCompliant>Pain in upper-left no.5</PatientCompliant>
                <Comments>...</Comments>
            </Observation>
        <DoctorCheckUp>
        <Xray date="12/11/2002" TechnicianName="Bob Cruise" id="TwelveNovXray">
            <Image type="UpperLeftOutside"
                ref="http://www.xlab.com/JJohnson/12/11/2002/id1234.jpeg"/>
            <Image type="UpperLeftInside"
                ref="http://www.xlab.com/JJohnson/12/11/2002/id1235.jpeg"/>
        </Xray>
    </Visit>
    <Insurance company="Clarica Health Care">
        <GroupPolicy>62738</GroupPolicy>
        <Type>...</Type>
    </Insurance>
    <Creditcard type="MasterCard">
        <Number>1234 5678 9128 2839</Number>
        <Expiry>06-06</Expiry>
        <Name>John Johnson</Name>
        <IssuedBy>BMO</IssuedBy>
    </CreditCard>
</PatientName>
```

Built-in Data Integrity Protection

C:\Work\Research\MS.NET\XML2.xml

```xml

<?xml version="1.0" encoding="utf-8" ?>
<Patient name="John Johnson" id="123-456-789">
    <Address>...</Address>
    <Visit date="11/11/2002" id="ElevnthNovemberRegular">
        <DoctorCheckUp>
            <Observation DoctorName="Michael Carter" id="ElNovObserv">
                <PatientCompliant>Pain in upper-left no.5</PatientCompliant>
                <Comments>...</Comments>
            </Observation>
            <Signature id="ElNovObservDoctorSig"
                xmlns="http://www.w3.org/2000/09/xmldsig#"/>
                <SignidInfo>
                    <CanonicalizationMethod
                        Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
                    <SignatureMethod
                        Algorithm="http://www.w3.org/2000/09/xmldsig#dsa-sha1"/>
                    <Reference URI="#ElNovObserv">
                        <Transforms>
                            <Transform
                                Algorithm="http://www.w3.org/TR/2001/REC-xml-c14n-20010315"/>
                        </Transforms>
                        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1">
                            <DigestValue>AE7jHg690hghg5ojPh79jhJkg478j88hd=</DigestValue>
                        </DigestMethod>
                    </Refernce>
                </SignedInfo>
                <SignatureValue>MSHjsu092ehd3981uqwjdlkj20=</SignatureValue>
            </Signature>
        <DoctorCheckUp>
</PatientName>
```

Built-in Data Confidentiality Protection

C:\Work\Research\MS.NET\XML3.xml

```xml

<?xml version="1.0" encoding="utf-8" ?>
<Patient name="John Johnson" id="123-456-789">
    <Address>...</Address>
    <Visit date="11/11/2002" id="ElevnthNovemberRegular">
        <DoctorCheckUp>
            ...
        <DoctorCheckUp>
        <Xray date="12/11/2002" TechnicianName="Bob Cruise" id="TwelveNovXray">
            ...
        </Xray>
    </Visit>
    <Insurance company="Clarica Health Care">
        <GroupPolicy>62738</GroupPolicy>
        <Type>...</Type>
    </Insurance>
    <Creditcard type="MasterCard">
        <EncryptedData xmlns="http://www.w3.org/2001/04/xmlenc#"
            Type="http://www.w3.org/2001/04/xmlenc#Element">
            <EncryptionMethod Algorithm="http://www.w3.org/2001/04/xmlenc#tripledes-cbc"/>
            <CipherData>
                <CipherValue>lkjsadloq=</CipherValue>
            </CipherData>
        </EncryptedData>
    </CreditCard>
</PatientName>
```

DOMINION VOTING SYSTEMS

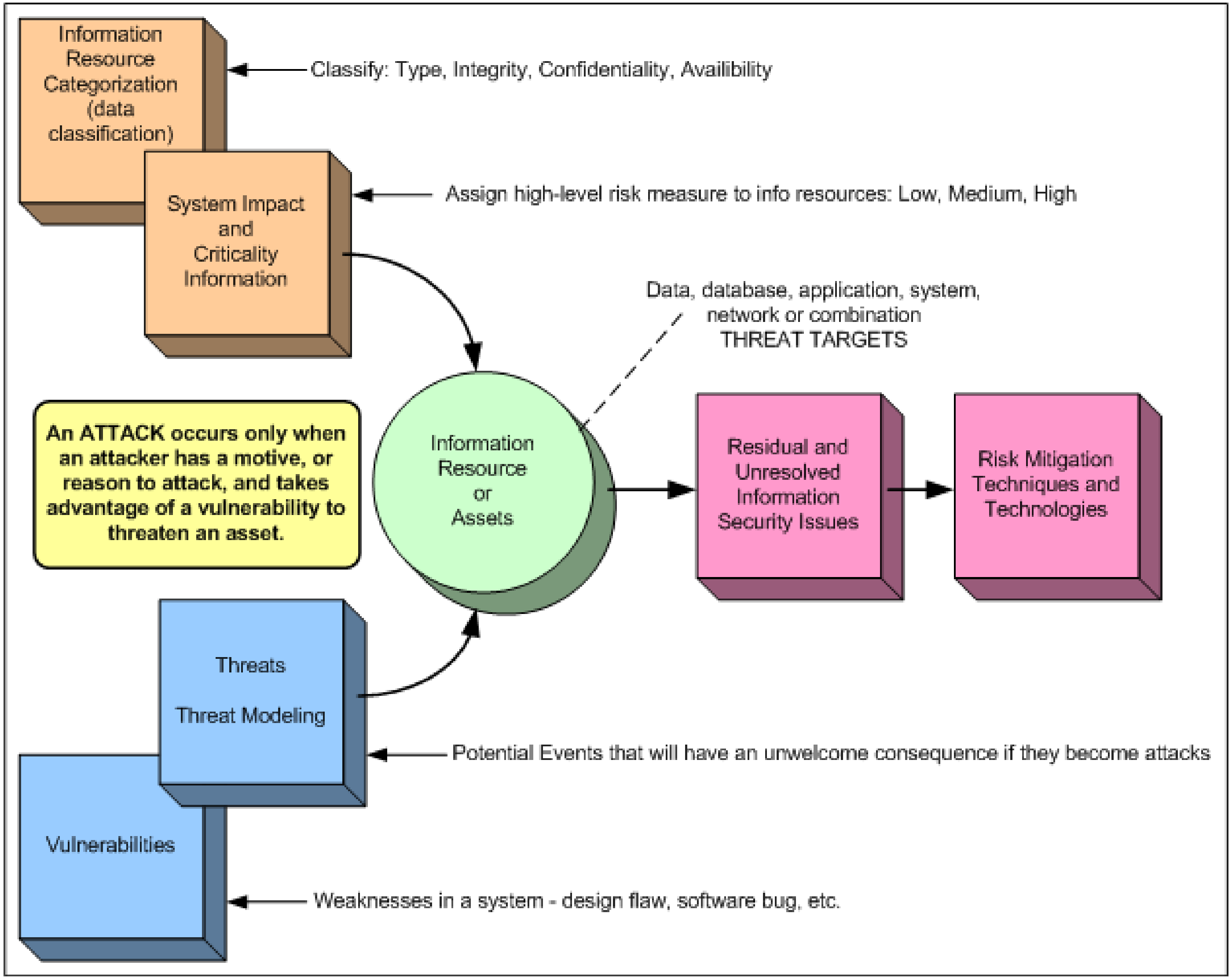
Info Security Risk Analysis – The Journey

Three ingredients must be present for an attack to occur:

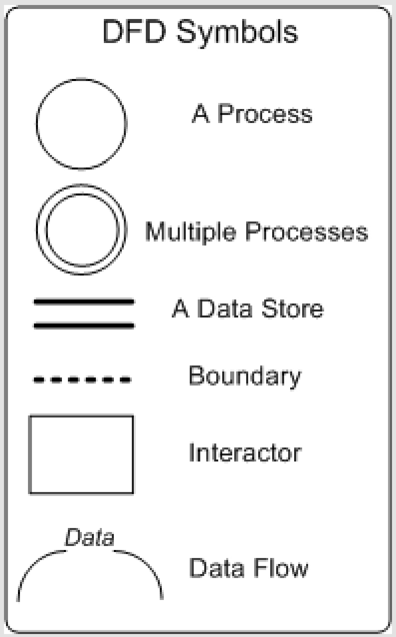
1. *Threats*
2. *Vulnerabilities*
3. *Assets*

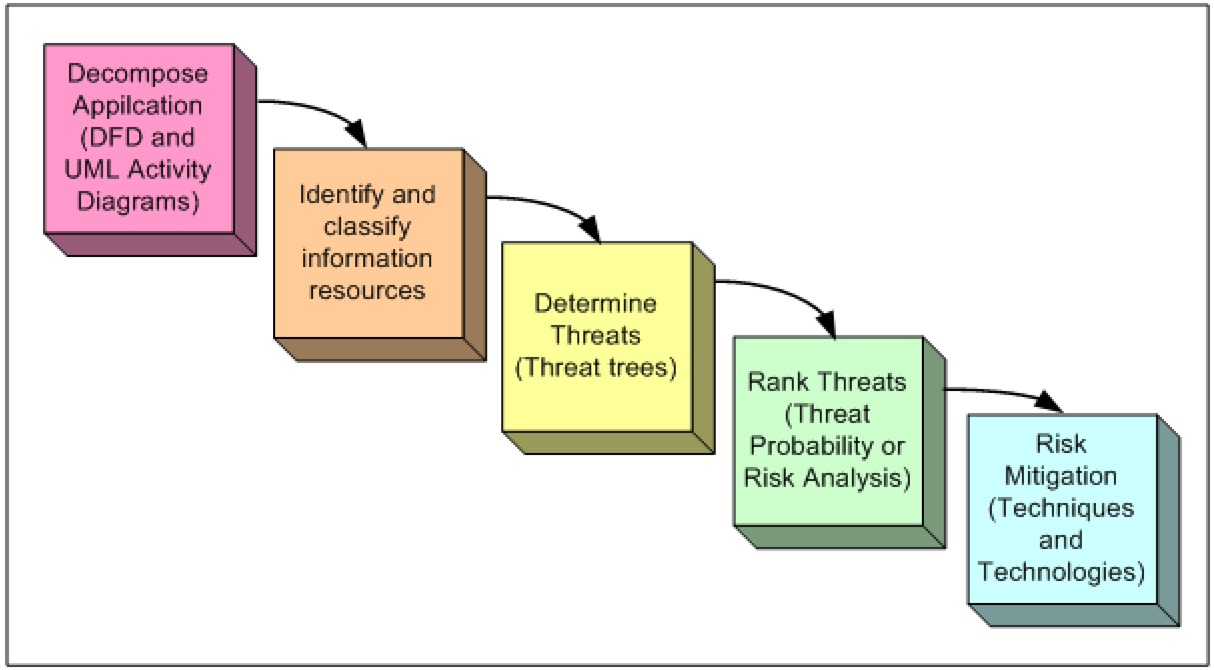
Take one of them away, and there will be no attack

Analogy – heat, oxygen and fuel are needed for fire



The Process of Threat Modeling





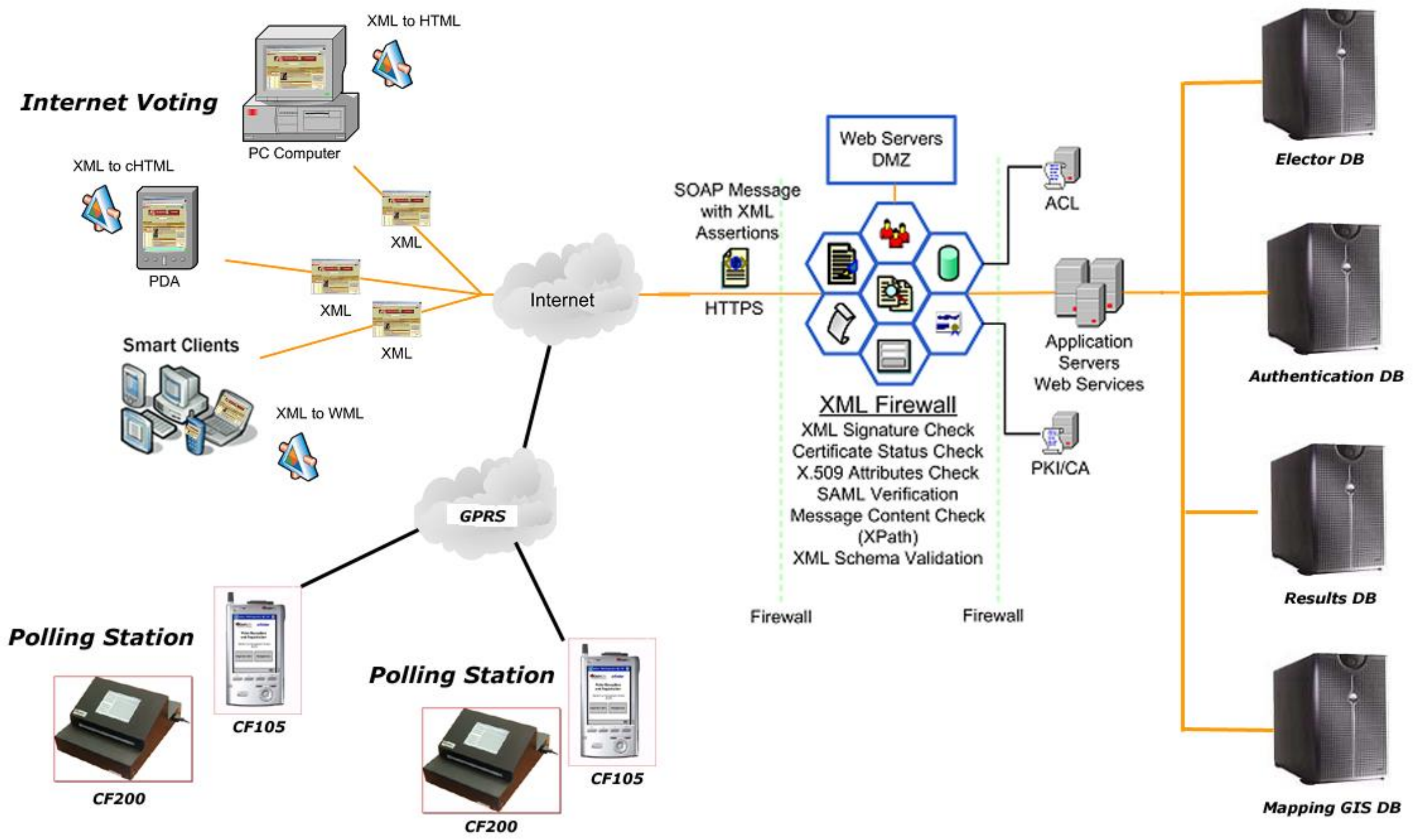
- *DFD – Data Flow Diagram*
- *DFDs focus on flow of data between processes, while UML Activity Diagrams focus on flow of control between processes.*

Decompose Application

Sample Application – Electronic Voting System

System Components:

- *Various DBs*
- *eVote Suite Applications*
- *eVote Internet*
- *CF105 – Voter Tracking and Registration*
- *CF200 – Electronic Voting Machines*
- *CF2000 – High-speed Central Count Voting Machines*
- *Communication Infrastructures*



DOMINION VOTING SYSTEMS

Decompose Application – Cont.

High-level Diagram for Internet Voting (small portion of it)

- This is Level-0 DFD Diagram

Interactors are:

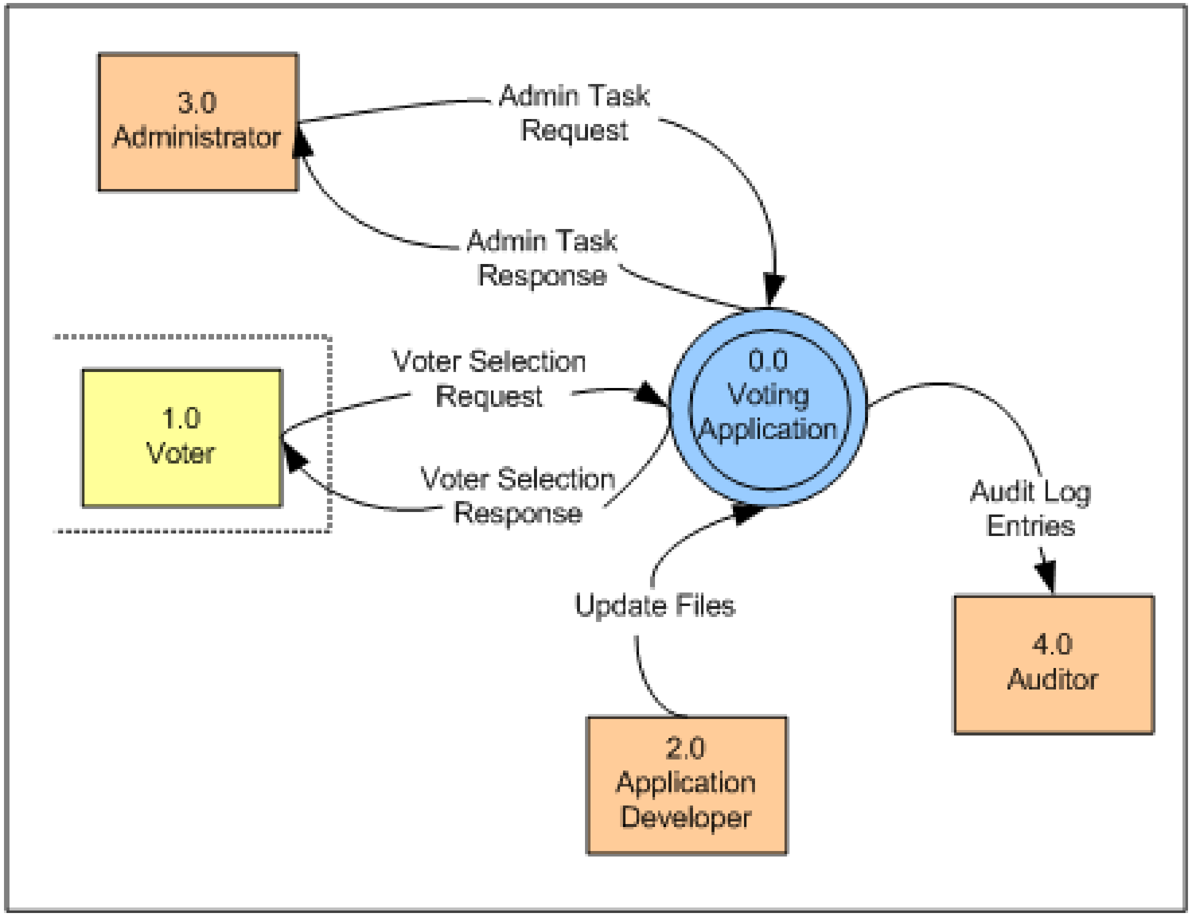
- Voters – external

- Administrators

- Application Developers

- Auditors

At this stage we only have high-level view of the system functionality



Decompose Application – Cont.

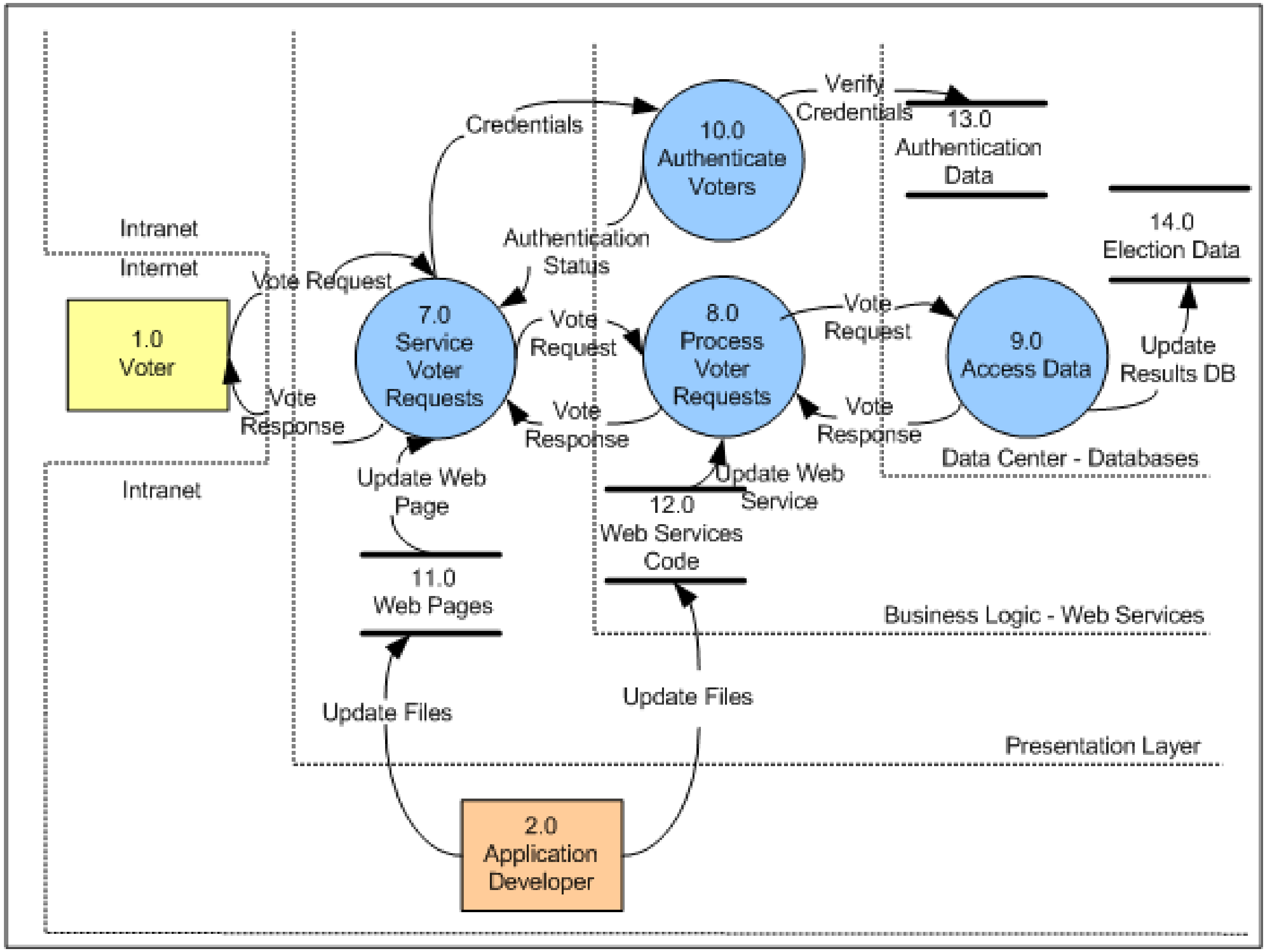
-This is Level-1 DFD Diagram

- We have better picture of processes, data flows and data stores at this view

- We should stop in decomposing the system when we determine exact usage scenarios of the system and how interactors use the system

- Be careful not to get into analysis paralysis

More Detailed Diagram for Internet Voting



Information Resources

Resources can be:

- Permanent or temporary data stores

- Computers

- Communication links and equipment

Identified Resources:

- *Server Computers – Web Server, Application Server, DB Servers*
- *Workstations and PCs – Voter PC and Developer Workstation*
- *Data Stores – Authentication DB, Results DB, source code store for Web pages and Web Services*
- *Communication Links – Internet links (wireline and wireless)*
- *Communication Links – Intranet links – LAN*

Classification Example:

➢ *Authentication DB:*

  - ❖ *Type – Highly-sensitive Information*
  - ❖ *Integrity – High*
  - ❖ *Confidentiality – High*
  - ❖ *Availability - High*

DOMINION VOTING SYSTEMS

Determine Threats

Other methods:

-OCTAVE

“Operationally Critical Threat, Asset and Vulnerability Evaluation“

Carnegie Mellon University

*Use **STRIDE** (Microsoft) methodology to categorize threats:*

- S – Spoofing Identity – allow an attacker to pose as another user or allow a rogue server to pose as a valid server – user or server authentication

- T – Tampering with Data – involves malicious modification of data – data integrity

- R – Repudiation – prevents denial of action

- I – Information Disclosure – involves the exposure of information to individuals who are not supposed to have access to it – data confidentiality

- D – Denial of Service – deny system or service access to valid user – service or system availability

- E – Elevation of Privilege – occurs when an unprivileged user gains privileged access to the system – user authorization

Threat Type	Processes	Data Stores	Interactors	Data Flows
S	Y	N	Y	N
T	Y	Y	N	Y
R	N	Y	Y	Y
I	Y	Y	N	Y
D	Y	Y	N	Y
E	Y	N	N	N

DOMINION VOTING SYSTEMS

Threat Tree – Example 1

Explanations:

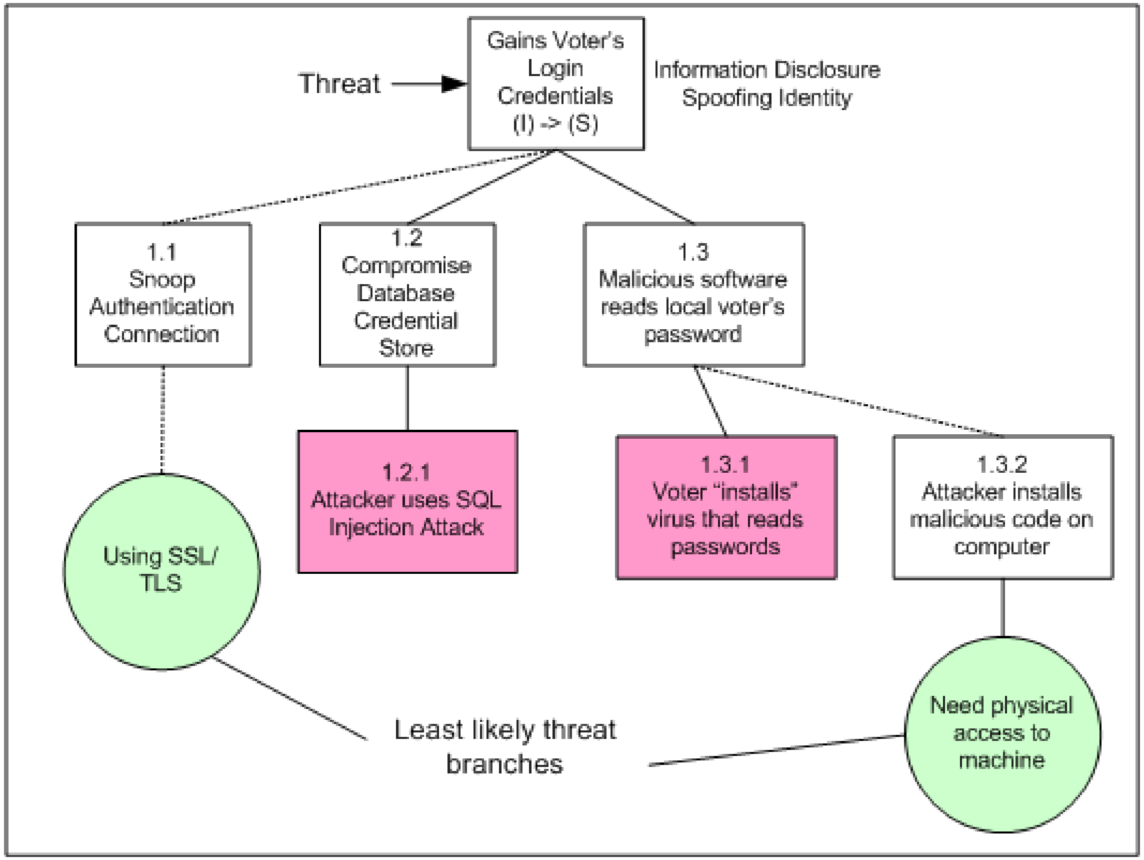
-Dotted line represents paths toward less likely scenarios

- Green circles denote possible mitigation technique

- Red boxes are scenarios with no obvious mitigation

Threat - Attacker gains voter authentication credentials

This is an example with multiple threat targets



Threat Tree – Example 2

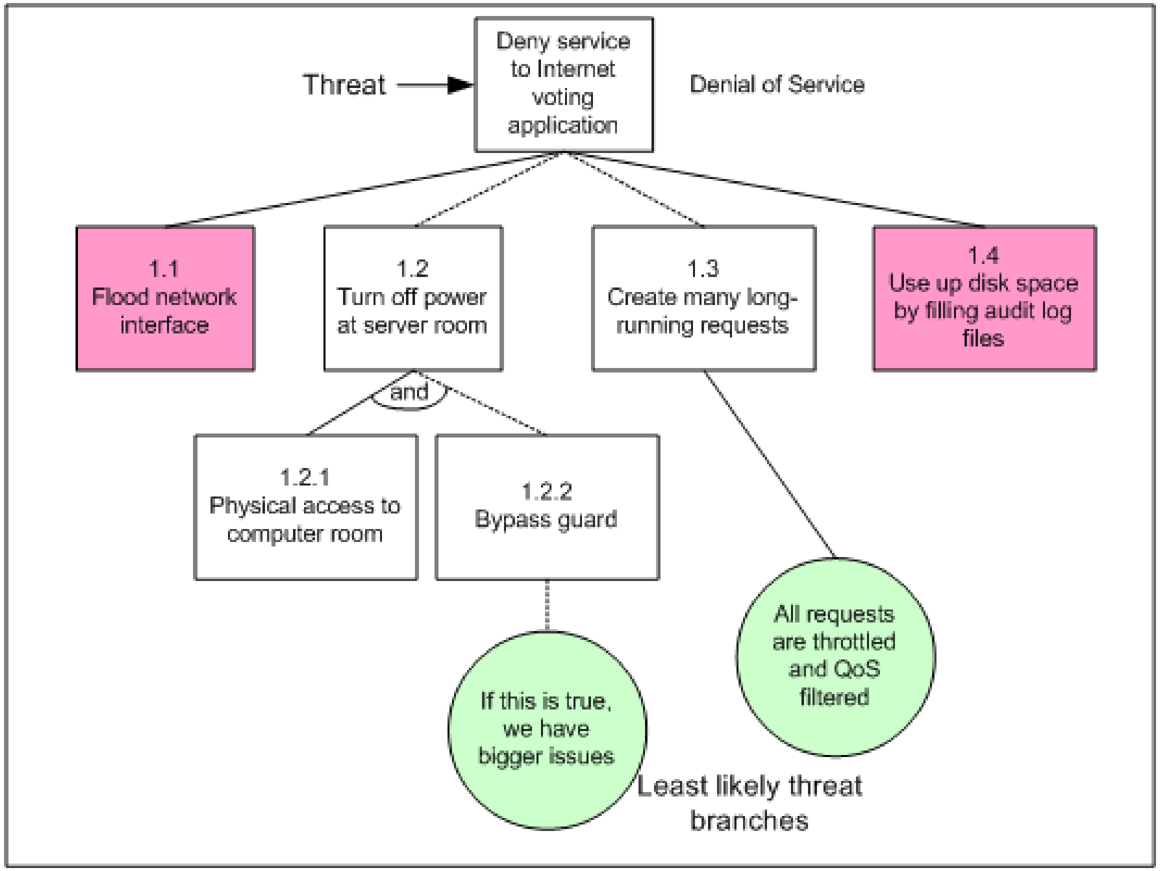
Explanations:

- Sometimes two or more events must happen (multiple vulnerabilities exploited) for an attack to be successful

- Helps in threat tree pruning if one scenario is mitigated

Threat - Attacker uses DoS or DDoS attacks to reduce availability of the system

Another multiple threat target example



DOMINION VOTING SYSTEMS

Rank Threats

Threat probability or risk in info security systems has lot of common with classical game theory – multiple players, each with his/her own motives and strategies

- *Try to calculate threat probability – risk*
- *A simple way:*

  Risk = Criticality x Likelihood of Occurrence

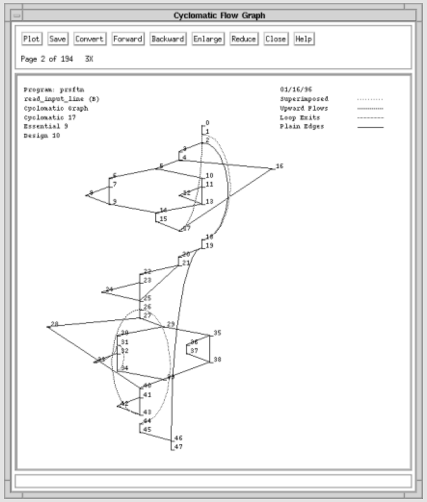
- *At DVS we use the following:*
  - *Cyclomatic software complexity measurements*
  - *Number of Affected Users*
  - *Damage Potential*
  - *Level of skill needed*
  - *Cost of attack*
  - *Reproducibility*
  - *Discoverability*
- *Assign values from 1 to 10 to each category (except software complexity)*
- *Quantitative risk value will be average of the above values*

DOMINION VOTING SYSTEMS

Rank Threats – Example 1

Software complexity factor calculates number of possible execution paths of a software module.

We want this factor to be below 10.



Threat - Attacker gains voter authentication credentials

Threat Target	In this example we will consider front end process for voter requests processing
Threat Category	Information Disclosure -> Spoofing of Identity
Software Complexity Factor	17
Risks	*Number of Affected Users: 10* *Damage Potential: 10* *Level of skill needed: 6* *Cost of attack: 9* *Reproducibility: 9* *Discoverability: 10*
Total Risk	*9*
Possible Mitigation	*Use regular expression to filter user input* *Use SSL/TSL to protect data traffic* *Consider user certificates for authentication*

Rank Threats – Example 2

DDoS program called Tribe Flood Network (TFN) was so potent that even one daemon attacking a Unix workstation disabled it to the point where it had to be rebooted

Communication equipment DoS issues - Majority of routers are very sensitive on fragmented TCP/IP packets

Threat - Attacker uses DoS or DDoS attacks to reduce availability of the system

Threat Target	In this example we will consider the whole electronic voting system and specifically Web Server as a threat target
Threat Category	Denial of Service
Software Complexity Factor	NA
Risks	*Number of Affected Users: 10* *Damage Potential: 6* *Level of skill needed: 6* *Cost of attack: 8* *Reproducibility: 8* *Discoverability: 10*
Total Risk	*8*
Possible Mitigation	*Use a firewall to drop certain IP packets* *Restrict resources used by anonymous users*

DOMINION VOTING SYSTEMS

Threat Mitigation Techniques

Some mitigation technologies are more secure than others, but also can be more expensive than others.

Always map mitigation technology to the corresponding threat based on information resource categorization and threat probability

There is no point of using strong encryption for publicly known information – phone numbers are one example.

Partial list of Threat Mitigation Techniques

Spoofing Identity	*Appropriate authentication* *Protect secret data* *Don't store secrets*
Tampering with Data	*Appropriate authorization* *Hashes* *Digital Signatures*
Repudiation	*Digital Signatures* *Timestamps* *Audit trails*
Information Disclosure	*Authorization* *Encryption*
Denial of Service	*Appropriate authentication and authorization* *Filtering, throttling* *Quality of Service*
Elevation of Privilege	*Run with least privilege*

DOMINION VOTING SYSTEMS

Internet Voting Big Security Problems

Other problems:

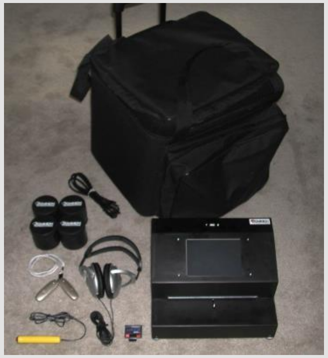
- Vote selling – the opportunity for voters to sell their vote

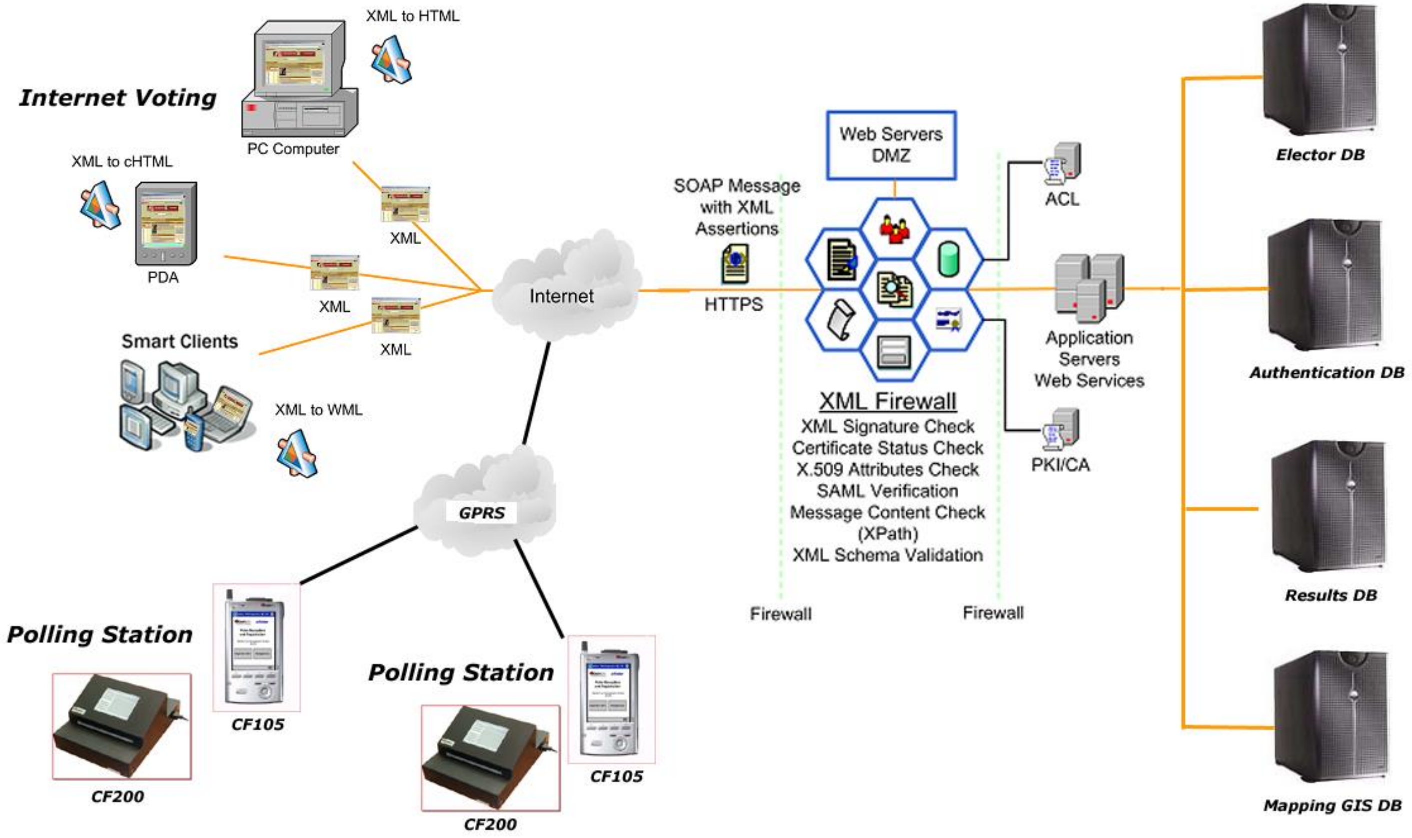
- Vote solicitation – the danger that outside of public polling station, it is much more difficult to control vote solicitation by political parties at the time of voting

- *Much easier to protect server side of the system, than home computers to be used for voting*
- *Malicious code is virtually limitless in the damage it can cause on a voting client – for example it can change the voter's vote regardless of encryption or authentication used*
- *Examples:*
  - *Backorfice 2000 – admin toolkit with full source code that runs in stealth mode. Can be used for remote administration with full control of the user's machine*
  - *CIH virus – time-bomb that can damage BIOS*
  - *Tampering with Proxy server configuration in web browsers*
- *There are several delivery mechanisms for malicious code – email (virus Bubbleboy activates in email client preview mode), operating systems and applications with security flaws, …*

DOMINION VOTING SYSTEMS

Where are we now?





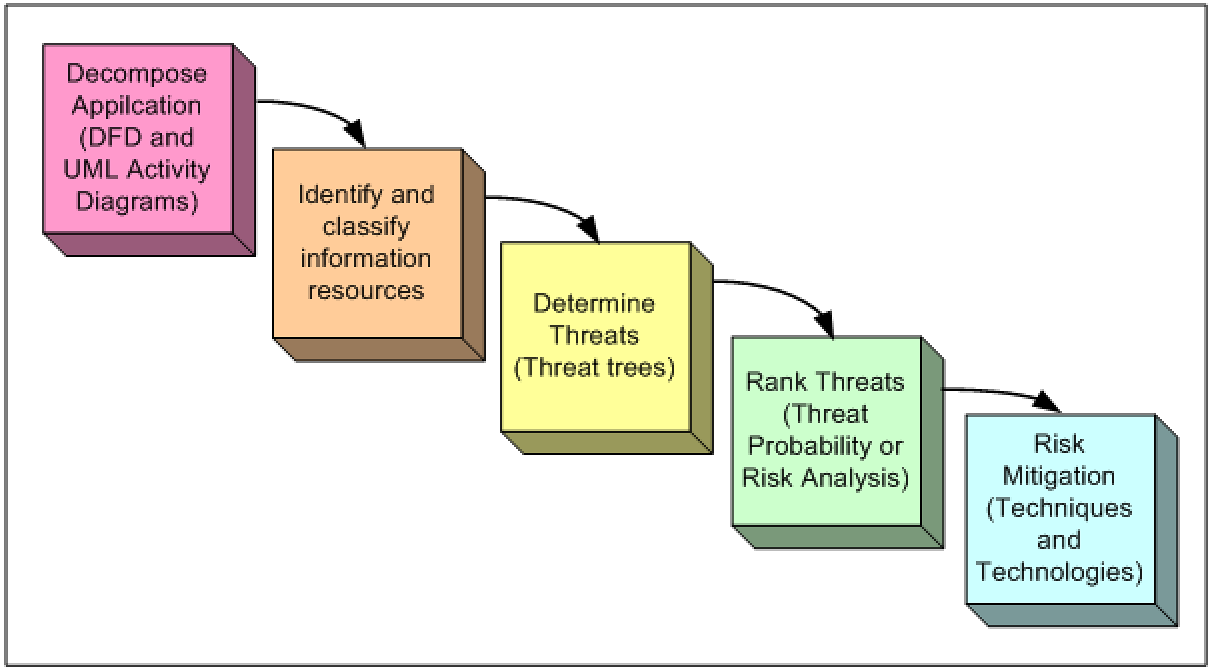
DOMINION VOTING SYSTEMS

Conclusions

Try to avoid Clausewitz syndrome (Carl von Clausewitz - German theoretician of war).

We have to recognize and implement in everyday development practice that application security is not just about firewalls and passwords. Application security is much more about the business context within which the application is implemented

- Before any decision on what security controls should be used for protection of information assets or system infrastructure, thorough risk analysis must be performed.
- Data sensitivity classification and threat modeling are two of the fundamental prerequisite steps needed for risk analysis, which in turn provide security basis of requirements engineering process.



DOMINION VOTING SYSTEMS

Questions and Answers

Goran Obradović

Director of Technology

Chief Information Security Officer

goran@dvscorp.com

DOMINION VOTING SYSTEMS

Identifying Threats & Threat Risk Modeling

고려대학교 (Korea Univ.)

사이버국방학과 · 정보보호대학원 (CIST)

보안성분석평가연구실 (Security Analysis aNd Evaluation Lab.)

김 승 주 (Seungjoo Kim)

(FB) www.fb.com/skim71 (Twitter) @skim71

고려대학교 정보보호대학원 KOREA UNIVERSITY