5

Introduction to Intrusion Tolerance

Fundamental Intrusion Tolerance Concepts

Some preliminary observations...

Should we bring the risk to zero?

- Let us talk about classical prevention/removal
  - *of the number and severity of the flaws of the system (vulnerabilities)*
  - *of the potential of the attacks it may be subjected to (threats)*
- We cannot make either arbitrarily low
  - too costly and infeasible
  - certain attacks come from the kind of service being deployed
  - certain vulnerabilities are attached to the design of the system proper
- ...and the question is: should we?
- can't we talk about acceptable risk?
- doesn't the hacker also incur in a cost of intruding??!!

And... can we?

- If we work on an all-or-nothing perspective, everytime we cannot assure something is completely secure, we have a problem of representation

(we don't know how to talk about "more or less secure" in formal terms)

What is Intrusion Tolerance?

- The tolerance paradigm in security:
  - Assumes that systems remain to a certain extent vulnerable
  - Assumes that attacks on components or sub-systems can happen and some will be successful
  - Ensures that the overall system nevertheless remains secure and operational, with a measurable probability
- In other words:
  - Faults--- malicious and other--- occur
  - They generate errors, i.e. component-level security compromises
  - Error processing mechanisms make sure that security failure is prevented

Trust and Trustworthiness
(support separation of concerns)

- Trust
- the accepted dependence of a component, on a set of properties (functional and/or non-functional) of another component, subsystem or system
  - a trusted component has a set of properties that are relied upon by another component (or components).
  - if A trusts B, then A accepts that a violation in those properties of B might compromise the correct operation of A
- Trustworthiness
- the measure in which a component, subsystem or system meets a set of properties (functional and/or non-functional)
  - trustworthiness of B measures the *coverage* of the trust of A

Trusted vs. Trustworthy

- *Thou shalt not trust non-trustworthy components!*
- B is Trustworthy in the measure its properties are met
  - ... and that coverage is never 1 in real systems...
- B should be Trusted only to the extent of its trustworthiness
  - trust may have several degrees, quantitatively or qualitatively
  - related not only with security-relat. properties (e.g., timeliness)
  - trust and trustworthiness lead to complementary aspects of the specification/design and implementation/verification process
- we should talk about trusted-trustworthy components

Intrusion Tolerance

terminology and concepts

Fault Models
Methodologies
Error processing
Fault treatment

Attacks, Vulnerabilities, Intrusions

- Intrusion
  - an externally induced, intentionally malicious, operational fault, causing an erroneous state in the system

an intrusion has two underlying causes:

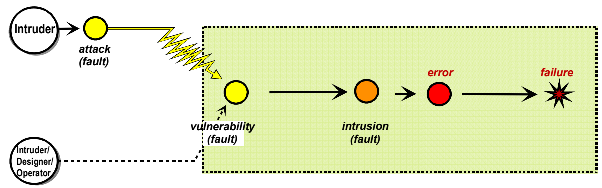
- Vulnerability
  - malicious or non-malicious weakness in a computing or comm's system that can be exploited with malicious intention
- Attack
  - malicious intentional fault introduced in a computing or comm's system, with the intent of exploiting a vulnerability in that system

interesting corolaries:

- without attacks, vulnerabilities are harmless
- without vulnerabilities, there cannot be successful attacks

Attack-Vulnerability-Intrusion composite fault model

Hence: attack + vulnerability → intrusion → error → failure
A specialization of the generic *"fault,error,failure"* sequence



AVI sequence : *attack + vulnerability → intrusion → error → failure*

Intrusion Tolerance

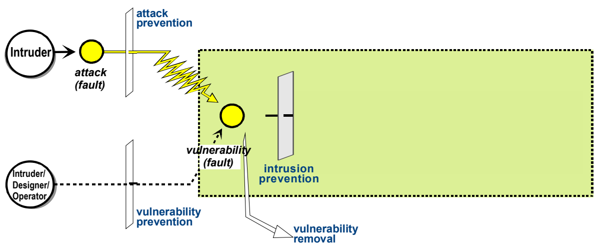
Fault Models
Methodologies
Error processing
Fault treatment

Achieving trustworthiness w.r.t. malicious faults
(the classical ways...)

- Attack prevention
  - Ensuring attacks do not take place against certain components
- Attack removal
  - Taking measures to discontinue attacks that took place
- Vulnerability prevention
  - Ensuring vulnerabilities do not develop in certain components
- Vulnerability removal
  - Eliminating vulnerabilities in certain components (e.g. bugs)

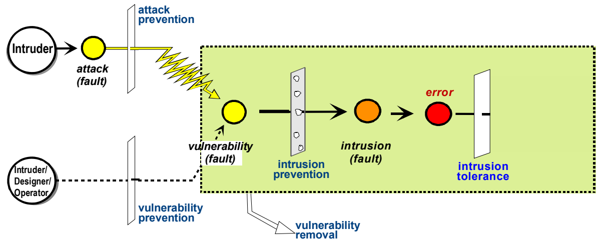
INTRUSION PREVENTION

Avoiding security failure
canonical track: intrusion prevention



➢ sequence : *attack + vulnerability→ intrusion→ failure*

Avoiding security failure
less canonical track: intrusion tolerance



➢ to be studied in this course ...

Intrusion Tolerance

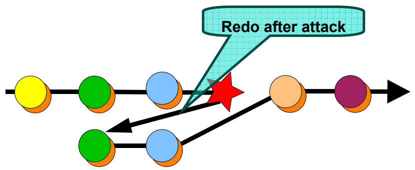
Fault Models
Methodologies
Error processing
Fault treatment

Processing the errors deriving from intrusions

- error detection
  - *detecting the error after it occurs aims at: confining it to avoid propagation; triggering error recovery mechanisms; triggering fault treatment mechanisms*
  - E.g.: modified files or messages; phony OS account; sniffer in operation; host flaky or crashing on logic bomb
- error recovery
  - *recovering from the error aims at: providing correct service despite the error*
  - E.g.: recovering from effects of intrusions

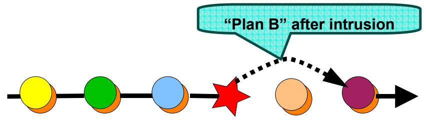
Processing the errors deriving from intrusions

- backward recovery:
  - *system goes back to a previous state known as correct and resumes*
  - system suffers DOS (denial of service) attack, and re-executes the corrupted operation
  - system detects corrupted files, pauses, reinstalls them, goes back
  - system detects corrupted message signature, discards, send nack



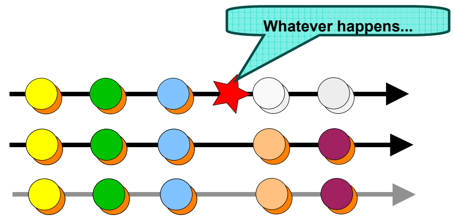
Processing the errors deriving from intrusions

- forward recovery:
  - *proceeds forward to state that ensures correct provision of service*
  - system detects intrusion, considers corrupted operations lost and increases level of security (threshold/quorums increase, key renewal)
  - system detects intrusion, moves to degraded but safer op mode



Processing the errors deriving from intrusions

- error masking
  - *redundancy allows providing correct service without noticeable glitch*
  - voting, Byzantine agreement; fragmentation-redundancy-scattering
  - sensor correlation (agreement on imprecise values)



Intrusion Detection

Classical methodologies
ID as error detection
ID as fault diagnosis

ID: Error detection or fault diagnosis?

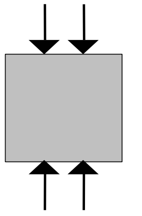
- classical IDS have two facets under intrusion tolerance
  - detecting errors as per the security policy specification
  - diagnosing faults as per the system fault model
- consider the following example:
  - *Organization A has an intranet with an extranet connected to the public Internet. It is fit with an IDS*
  - the IDS detects a port scan against an internal host, coming from the intranet
  - the IDS detects a port scan against one of the extranet hosts, coming from the Internet
  - *what is the difference?*

A biologically inspired metaphor of intrusion tolerance

Courtesy Christian Cachin, MAFTIA consortium

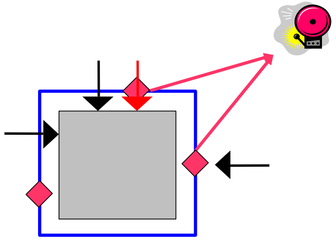
Computer system under attack

- no flaws, no vulnerabilities



Intrusion detection

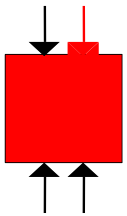
- Sensors for different attacks



Sensor

Computer system under attack

- with vulnerabilities and
- successful attack

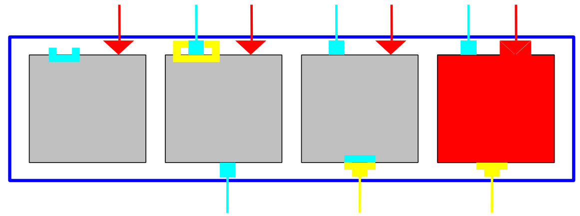


Vulnerability

Attack that exploits the vulnerability

Intrusion Tolerance

- with replicated and diverse structure
  - replicas have different vulnerabilities
  - majority remains intact

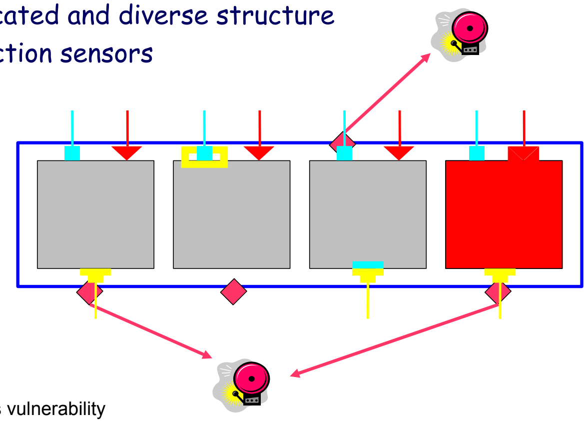


Vulnerability

Attack that exploits the vulnerability

Intrusion Tolerance and Detection combined

- with replicated and diverse structure
- with detection sensors

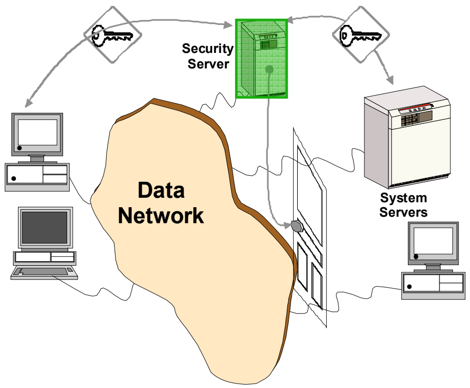


Vulnerability

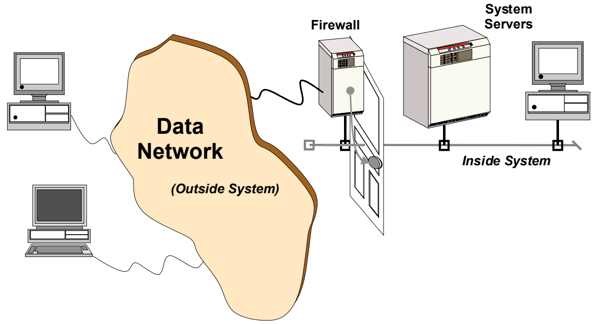
Attack that exploits vulnerability

Example Intrusion-Tolerant Networks and Architectures

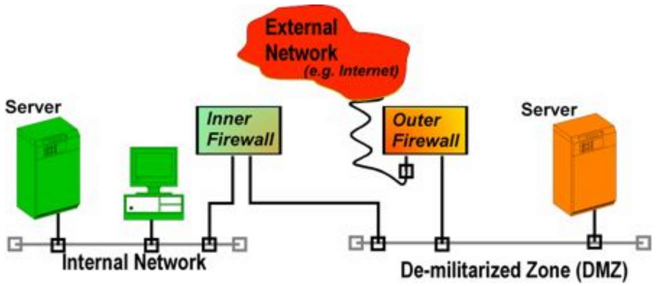
Trusted-Third-Party Security Server



Intrusion-Prevention Firewall

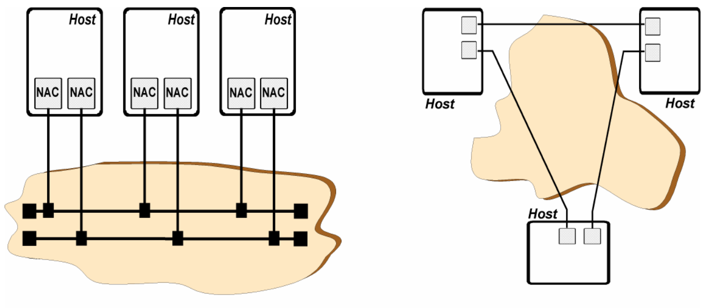


Firewalling



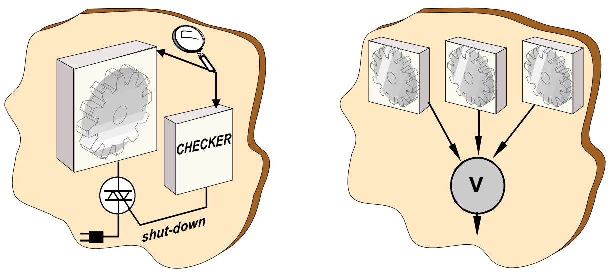
- Intrusion prevention device: prevents attacks on inside machines
- Coverage: semantics of firewall functions, resilience of bastions
- End-to-end problem: are all internal network guys good?

Intrusion-Masking Redundant Networks



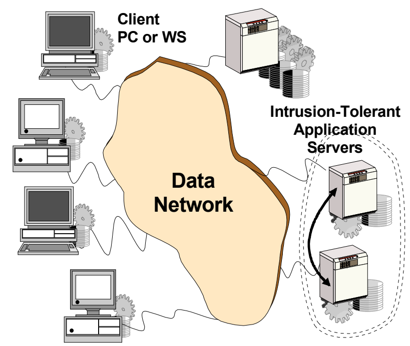
(a) (b)

Intrusion Detection and Masking in Processing

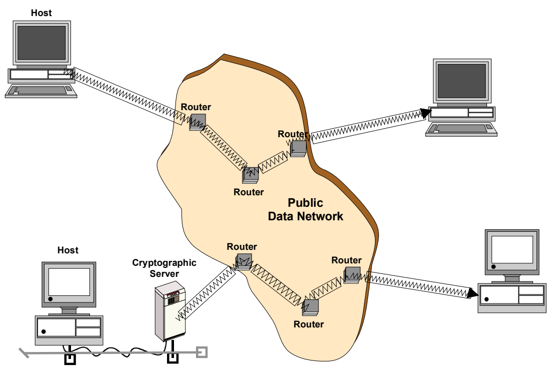


(a) (b)

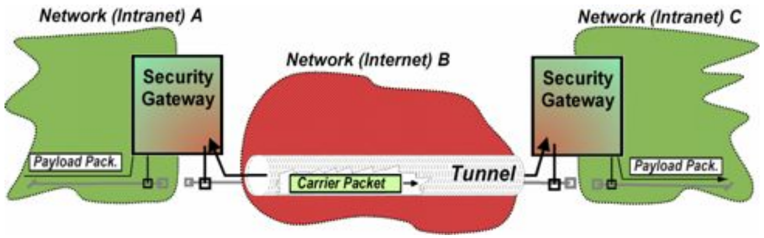
Client-Server with Intrusion Tolerant Servers



Intrusion-Prevention Secure Circuits

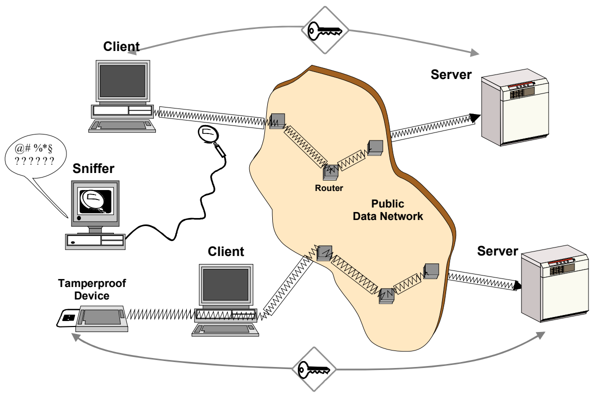


Tunnelling, secure channels



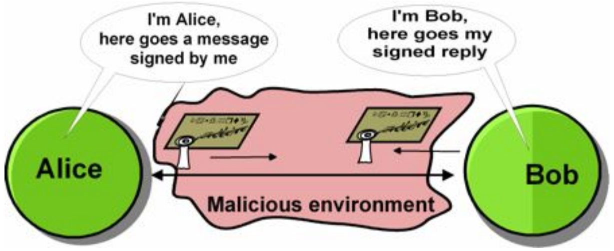
- Intrusion prevention device: enforces confidentiality, integrity (authenticity)
- Coverage: tunnelling method, resilience of gateway
- End-to-end problem: are all intranet guys good?

Secure Remote Operations



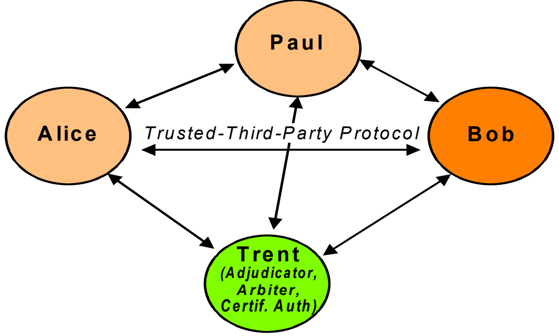
Other Example Intrusion-Tolerance mechanisms

Authentication, signatures, MACs



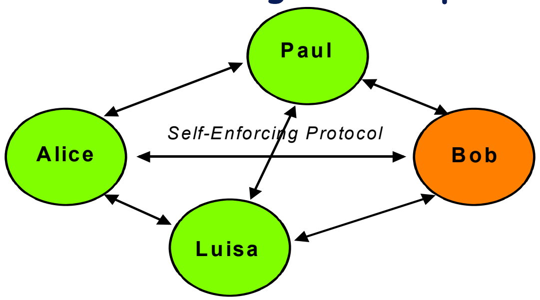
- Intrusion prevention device: enforces authenticity, integrity
- Coverage: signature/authentication method
- End-to-end problem: who am I authenticating? me or my PC?

Trusted Third Party (TTP) protocols



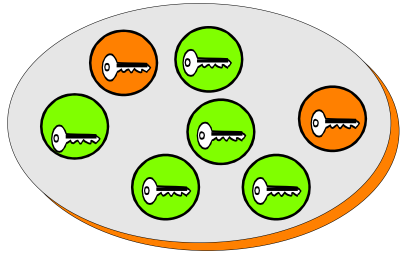
- Intrusion tolerance device: error processing/masking
- Coverage: semantics of protocol functions, underlying model assumptions, resilience of TTP

Communication and agreement protocols



- Intrusion tolerance device: error processing or masking (3f+1, 2f+1, f+2)
- Coverage: semantics of protocol functions, underlying model assumptions

Threshold cryptography



- Intrusion tolerance device: error processing/masking (f+1 out of n)
- Coverage: crypto semantics, brute force resilience, underlying model assumptions