

Recognising and Responding to the Insider Threat

Clive Blackwell
Computing and Communication Technologies
Faculty of Technology, Design and Environment
Oxford Brookes University
Wheatley Campus, Wheatley,
Oxford OX33 1HX
cblackwell@brookes.ac.uk

Cyberpatterns 2012

Proceedings of the First International Workshop on
Cyberpatterns: Unifying Design Patterns with
Security, Attack and Forensic Patterns

9-10 July 2012

The Cosener's House, Abingdon, UK

Organized and Sponsored by

Oxford Brookes University

Sponsored by

SOPHOS

In association with

**BCS Information Security
Specialist Group**

**BCS Formal Aspects of Computing
Specialist Group**

**BCS Cybercrime Forensics
Specialist Group**

Cyberpatterns

[http://tech.brookes.ac.uk/
CyberPatterns2012](http://tech.brookes.ac.uk/CyberPatterns2012)

Clive Blackwell
Computing and Communication Technologies
Faculty of Technology, Design and Environment
Oxford Brookes University
Wheatley Campus, Wheatley,
Oxford OX33 1HX
cblackwell@brookes.ac.uk

7-layer OSI network model

Strong influence on my multilayered model

Paths are conceptually horizontal

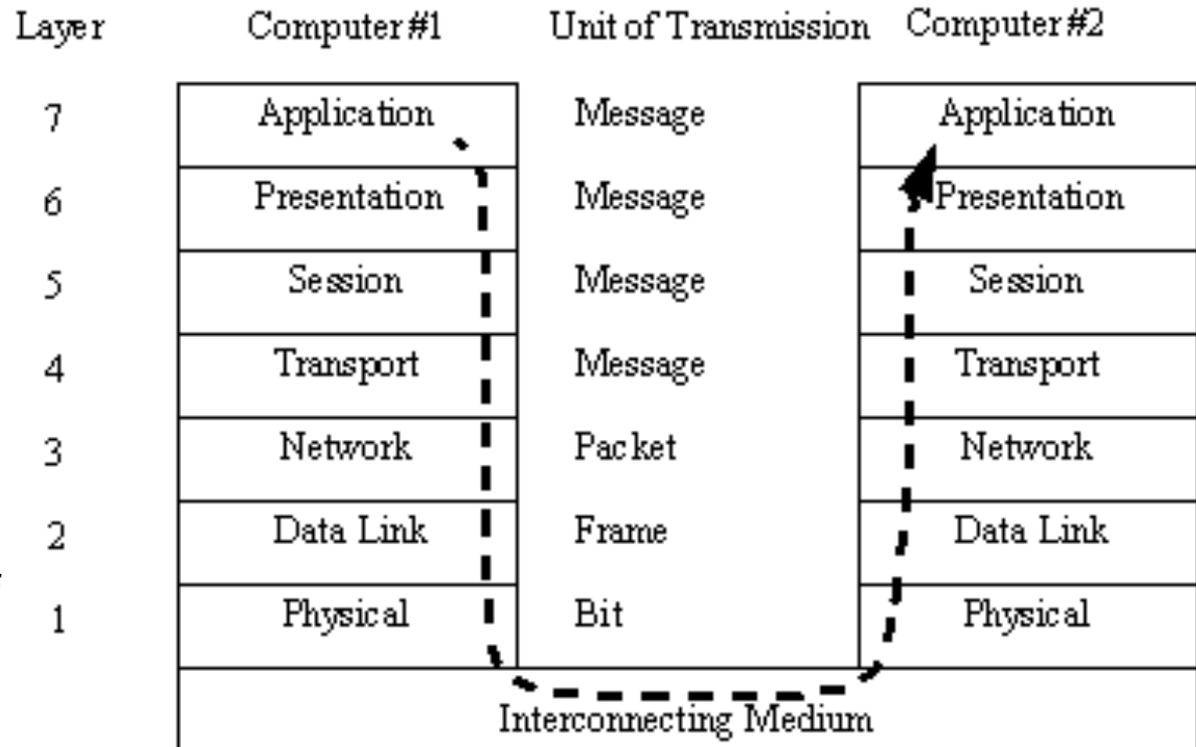
Pass down the levels and physical transmission in reality

My model explicitly includes people and the physical world

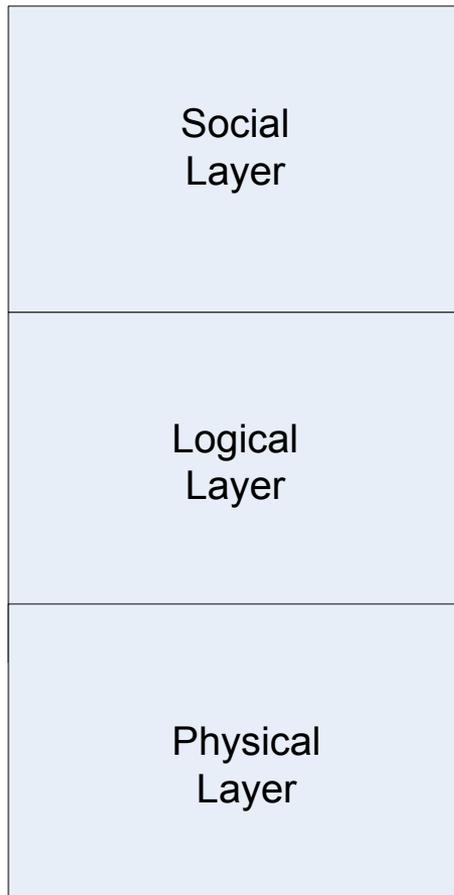
People here modelled by application level proxies

Physical medium is out of scope

We considers action and state as well as communication

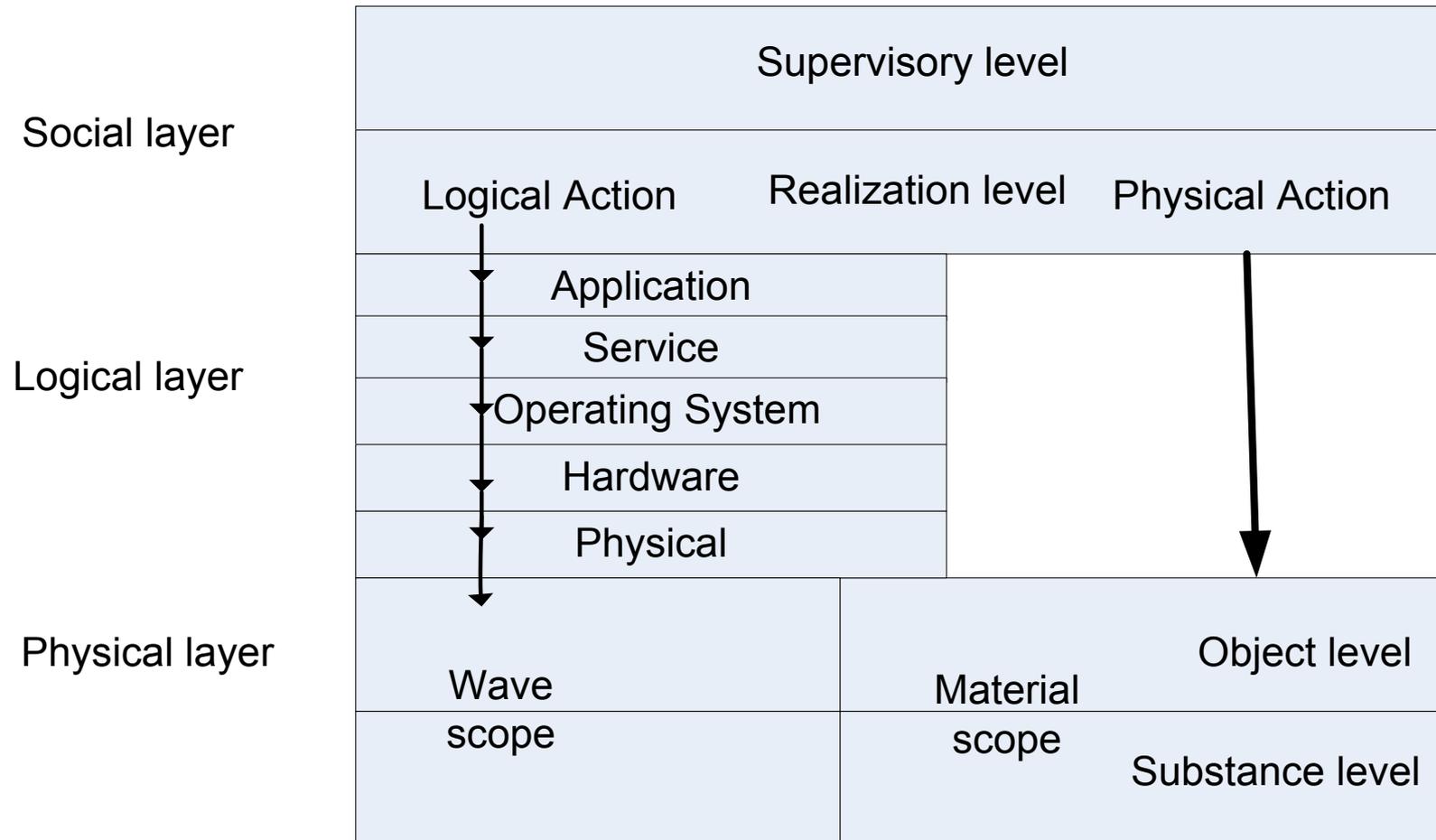


The Layered Security Model

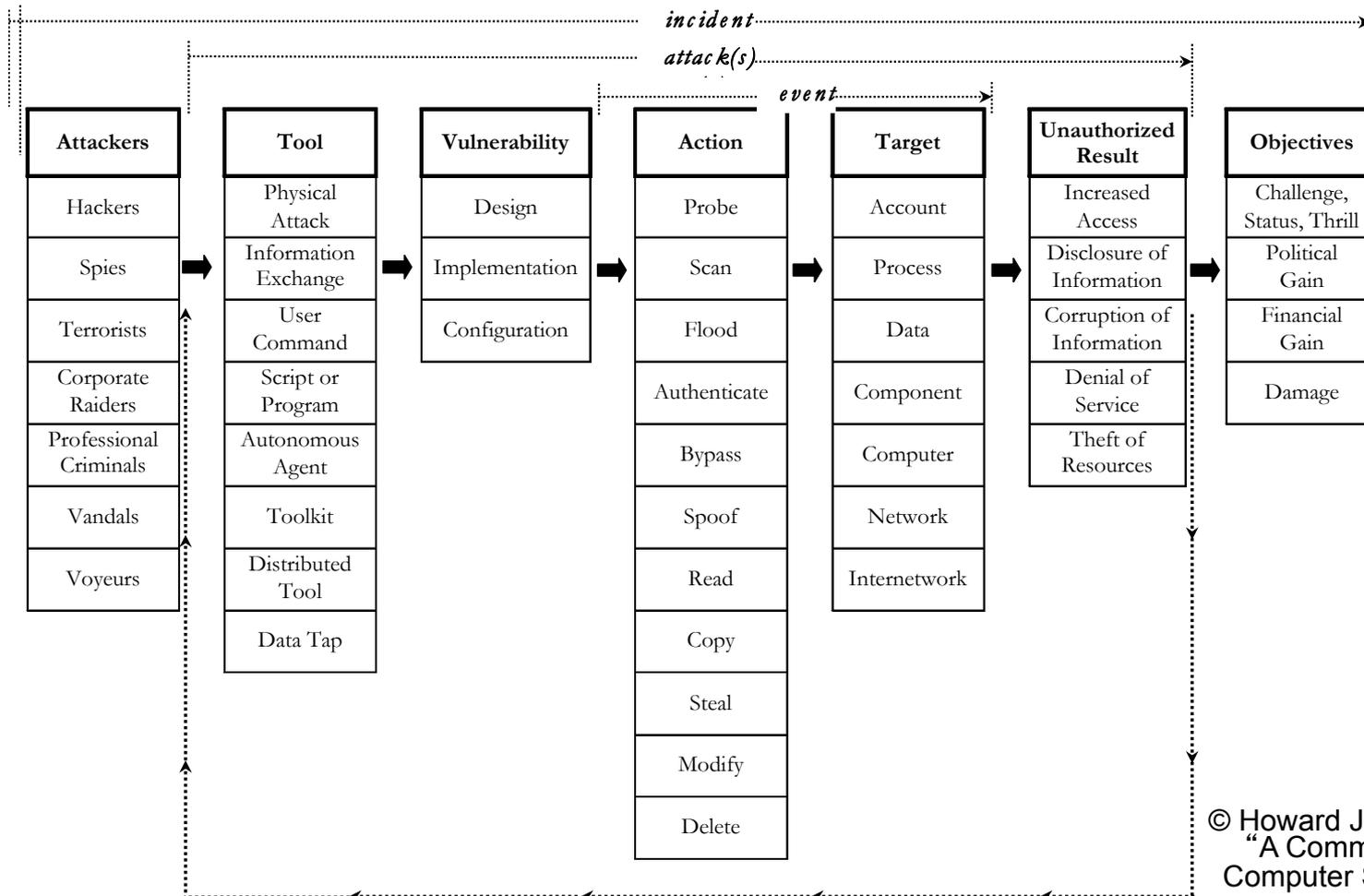


- Have a simplified three-layer model
 - Add sub-layers and horizontal scope to recover the greater number of layers in other models
- Social layer at the top includes people and organisations along with their goals and intentions
 - Legal, organisational, economic, philosophical, political, sociological, and psychological aspects
- Logical layer in the middle contains computers, networks and software
 - Has multiple sublevels to recapture the layers of the other mainly logical models
- Physical layer at the bottom represents the physical existence of all entities in world
 - Contains tangible objects including buildings, equipment, paper documents and computers
 - Also contains physical phenomena such as electromagnetic radiation (radio waves), electricity and magnetism

The layered model



Howard's Computer Security Incident Taxonomy



© Howard JD and Longstaff TA, "A Common Language for Computer Security Incidents"

Howard and Longstaff's security classification

- John Howard invented a classification for network security incidents in his thesis (Carnegie-Mellon 1997)
- Shows the different types of entity involved in incidents, their characteristics and the relationships between them
 - Offers no advice for organising defence
- Seafood menu according to Longstaff
 - Set of independent choices with one item selected from each column
- A useful informative conceptual model that we extend
- Categories are attacker, tool, vulnerability, action, target, unauthorised result and objectives
- *Attacker* uses a *Tool* to exploit a *Vulnerability* performing an *Action* on a *Target* to cause an *Unauthorised result* to meet its *Objectives*

JD Howard and TA Longstaff, *A Common Language for Computer Security Incidents*, Sandia National Labs, 1998, at www.sandia.gov.

Extension of Howard's model

- A comprehensive set of categories and their subdivisions
- Incidents described at all layers
 - Including social and physical processes, states and properties
- Inclusion of levels aids holistic incident analysis
- Incident inhabits a connected conceptual space
 - Each entity has a location, and moves and interacts within this space
- Can analyse incident paths within this space
 - Incident progresses from left to right
- Differentiate between incident and stage aspects
 - Both incident and its stages has access, use and effect elements
- Differentiate between actual and latent effects
 - Stage aspects often just defeat STRIDE in the control domain (means)
 - Ultimate incident effects are real, including psychological effects (ends)
- Have a corresponding defensive classification
 - Separate incident from defensive response and analysis
 - Vulnerability in Howard's classification part of defensive taxonomy here

Incident table

Incident or stage entity or attribute	Perpetrator	Method	Action	Stage Actor or Agent	Method	Action	Target	Immediate effect	Ultimate effect	Ultimate target
Investigative questions	Ultimate Who	Complete with what	Complete how	Stage who	Stage with what	Stage how	Stage to what	Stage what	Ultimate What	Ultimate to what
Supervisory										
Social level										
Realization										
Application										
Service										
Operating system										
Logical level										
Hardware										
Physical										
Object										
Physical level										
Substance										

Zachman framework for designing enterprise IT architecture

	Why	How	What	Who	Where	When
Contextual	Goal List	Process List	Material List	Organizational Unit & Role List	Geographical Locations List	Event List
Conceptual	Goal Relationship	Process Model	Entity Relationship Model	Organizational Unit & Role Rel. Model	Locations Model	Event Model
Logical	Rules Diagram	Process Diagram	Data Model Diagram	Role relationship Diagram	Locations Diagram	Event Diagram
Physical	Rules Specification	Process Function Specification	Data Entity Specification	Role Specification	Location Specification	Event Specification
Detailed	Rules Details	Process Details	Data Details	Role Details	Location details	Event Details

Adapting and extending Zachman's questions

- Zachman's 2nd dimension to his model poses 6 questions to describe different system aspects
- Who, what, when, where, how and why
 - Answered by Zachman for each of the five levels, leading to a five by six grid
- Use as incident questions to analyze incident progression to guide analysis and response
 - Answered within my conceptual architectural space
- Each entity in my model helps to answer one question
 - Perpetrator (who) performs an action (how) to cause an effect (what) for a reason (why) at a particular location (where/when)
- Answer questions for the incident and for each stage
 - Innocent party may act in some stages of social engineering attack

Adapting and extending Zachman's questions

- Each entity in my model helps to answer one question
- Add six additional questions
 - 'with what' (tool or technique), 'to what' (target)
 - 'over what' (topology), and 'to whom' (victim)
- Abilities should be considered as well
 - 'know what' (knowledge) and 'know how' (skills)
- Explicit incident division into stages with specific objectives
 - Relate incident question to each entity
- In an active incident *stage*, the *actor* (who) uses a *method* (with what) to perform an *action* (how) that executes a *threat* to exploit a *vulnerability* with an *immediate effect* (what) on a *target* (to what)
 - Incident has a social-level goal (why) met by the *ultimate effect* (what), whereas immediate stage effect may have no direct impact
- Occurs within a conceptual space (where and when)
 - Includes location, communication and movement at all levels
 - Includes underlying physical or logical medium (over what) and intrinsic form of entities

Incident table

Incident or stage entity or attribute	Perpetrator	Method	Action	Stage Actor or Agent	Method	Action	Target	Immediate effect	Ultimate effect	Ultimate target
Investigative questions	Ultimate Who	Complete with what	Complete how	Stage who	Stage with what	Stage how	Stage to what	Stage what	Ultimate What	Ultimate to what
Supervisory										
Social level										
Realization										
Application										
Service										
Operating system										
Logical level										
Hardware										
Physical										
Object										
Physical level										
Substance										

iAssemble case study

The iAssemble incident is a fictional case study developed and used for training purposes by CERT and SEI that is representative of the many real cases of insider sabotage from their insider threat research

We show a small number of possible sabotage attacks by Ian Archer, a disgruntled iAssemble employee, which would be greatly extended in a realistic analysis

We discuss the corresponding defenses briefly, indicating defensive actions that may have been successful against each of these attack vectors to provide comprehensive defense-in-depth at all levels

AP Moore, DM Cappelli, RF Trzeciak, *The “Big Picture” of Insider IT Sabotage Across U.S. Critical Infrastructures*, TR CMU/SEI-2008-TR-009, Software Engineering Institute, Carnegie Mellon University, 2008.

Caveat – Incidents investigated by CERT may not be representative of usual sabotage incidents

Comparison of Logical Insider ↓ Incident Types

Attack type	Insider IT Sabotage	Insider Theft or Modification of Information for Financial Gain	Insider Theft of Information for Business Advantage
Percentage of crimes in CERT's case database	45%	44%	14%
Current or former employee?	Former	Current	Current
Type of position	Technical (e.g. system administrators or database administrators)	Non-technical, low-level positions with access to confidential or sensitive information (e.g. data entry, customer service)	Technical (71%) - scientists, programmers, engineers Sales (29%)
Gender	Male	Fairly equally split between male and female	Male
Target	Network, systems, or data	Personally Identifiable Information or Customer Information	Intellectual Property (trade secrets) – 71% Customer Information – 33%
Access used	Unauthorized access	Authorized access	Authorized access
When	Outside normal working hours	During normal working hours	During normal working hours
Where	Remote access	At work	At work
Recruited by outsiders	None	Half recruited for theft; less than one third recruited for modification	Less than one fourth
Collusion	None	Almost half colluded with another insider in modification cases; 2/3 colluded with outsiders in theft cases	Almost half colluded with at least one insider; half acted alone

© D Cappelli, A Moore, TJ Shimeall and R Trzeciak, “Common sense guide to prevention and detection of insider threats, version 3.1”, CERT (2009)

Damaging incidents

- Most sabotage is carried out by employees with a personal grudge against the organisation
- Logical attacks are usually perpetrated by technical staff
 - For example, systems administrators with privileged access
- Physical damage can be carried out by any employee
- Many saboteurs have left, but retain partial access
 - Have knowledge of system weaknesses
 - Not all rights may have been removed
 - Often use unauthorised access even if they retain their privileges
- A degree of sophistication and premeditation is often involved in technical attacks
 - Use remote access, malware such as logic bombs, backdoors, hidden or compromised accounts
 - May not be typical in general (possible skewed sample)

Damaging incidents (2)

- Objective is to cause damage to the organisation for psychological satisfaction for some perceived wrong
 - Disciplined or terminated, no salary increase or bonus, passed over for promotion or demoted, or given more work
- Immediate purpose is to destroy or damage the integrity and availability of physical or logical resources
 - Physical – people, buildings, machinery, controllers, computers
 - Logical – control systems, computers, programs and data
- Positive psychological effect caused by negative organisation effect
- Lower layer immediate effects intend to cause significant ultimate damage to the organisation (means)
- Ultimate impact includes lost production, lost business, loss of reputation and recovery costs (ends)
- Directed attacks against single point of failure hard to overcome
 - Destroying crucial systems and data

Insider threat from sabotage

Perpetrator (who) and Motivation (why)	Stage agent (who)	Reason (why)	Method (with what)	Action (how)	Target (to what)	Immediate Effect (what)	Ultimate effect (what)
Disgruntled former employee, Ian Archer. Motivation is psychological satisfaction from revenge for perceived mistreatment	Archer. 1a The targets unwittingly help to give unauthorized access to Archer 1b	To gain system access	Social engineering ♣ ∞ 1a alternative paths using email or phone (arrow extends to the physical row)	Persuade or trick target to act incorrectly by giving access, setting up accounts, or giving out passwords	Security guards, system administrators, colleagues	Unauthorized physical, or logical access (via a compromised account)	Inability to produce computers → Failure to satisfy contracts → Financial losses → Reduced reputation, lost customers, lowered share price
Social	1a						
Logical	1b Archer using his account	To gain hidden logical access after termination and avoid accountability	Misuse authorized authority using own account to issue commands	Logically authorized (but prohibited by policy at the social level) commands to set up a backdoor	Network access to system	Gain unauthorized remote logical access after termination	
Logical Remote	1c Archer using a compromised account and remote access	To install malware to maintain indirect system access and avoid accountability	Use of compromised account and remote access to issue commands	Unauthorized commands to install logic bomb	Operating system of computer holding production software	Installation of logic bomb <i>Backup loss</i>	
Logical Local	2/3 Logic bomb	To cause immediate damage to software → Ultimate goal to damage computer production	Privileged software misuses host system	Issue damaging commands to destroy files and software → May also cover tracks by deleting log files	Software on production control system and backups → Production processes Log files ♠	Unavailable production software → Lost computer production	
Physical	Archer <i>Physical attack goes from left to right cell to cell same as logical attack</i>	Render critical assets unavailable → Stop production line	Misuse allowed access to physically interfere with equipment and resources before leaving, illegitimate access after termination	Physical damage and destruction, theft, encryption	Software, backups, production computers and other essential equipment Logging devices	Damaged or unavailable systems and resources → Lost computer production	

Table explanation

- Archer is a disgruntled former employee that targets the organisation for his perceived mistreatment
- Table has different columns from the earlier one
 - This is an earlier version, but analysis remains the same though
- Top row has cells for the perpetrator, motivation, ultimate effect and ultimate target (not shown)
 - Only have meaning at the social layer
- Incident consists of three stages of access, use and effect
 - Marked 1, 2 and 3 respectively with a, b, c for different stages of the same type
- Causes on the left, actions in the middle and effects on the right
- There will be at least one of each stage in a successful incident, but there could be more
 - Archer's incident involved multiple access stages
- Archer acts at one step removed at each stage
 - Used remote access (physical) and a compromised account (logical) to plant a logic bomb (temporal) to maintain indirect access

Table explanation (2)

- Incident progresses from left to right in multiple stages
- Three active stages of access, misuse and effect
- Misuse and effect may be contemporaneous in sabotage
 - Misuse often causes damage that is also the intended effect
- Progression of each stage is from left to right
 - Then back to left to use newly acquired powers in the next stage
- 1st stage in sabotage is often to gain additional privileges
 - To avoid responsibility or interfere with other resources to stop recovery (eg destroy backups)
- Immediate effect on confidentiality, integrity and availability is the means to achieve social-level ends
- Ultimate motivation and effect at social level
 - Inability to produce goods, failure to satisfy contracts, financial losses, reduced reputation, lost customers, lowered share price
 - Ultimate effects satisfy Archer's psychological motivation

Access stage

- Uses authorised access at first, so not recognised as an attack
 - Forward arrow from the perpetrator to use logical (shown 1b) access
 - Physical access could also be used (arrow to the bottom row omitted)
- Immediate effects of the first stage gains unauthorised access
 - Moves within system or defeats controls by left to right movement
 - Backwards line marked ♦ starting in the top row shows immediate effect of acquiring the password to a colleague's account
 - Used in the logical row 1c to install the logic bomb
- Top row shows that Archer got a colleague to share their password, ostensibly to ease the performance of a legitimate task
 - Maintains access as passwords were not changed when he left
 - Can then follow colleague's conceptual access path
- Could also launch a social engineering attack
 - Trick a guard into giving unauthorised physical access after termination
 - Masquerade as another employee to get their password reset
- Sending email requesting password reset using another employee's email account shown by the detour to the logical level marked ♣
 - Email channel is similar to virtual communication of OSI model

Access stage (2)

- Archer needs both logical access and physical connectivity
- Archer ensured he had remote access, which he needed for a logical attack from outside after termination
 - Would otherwise need physical access by getting past the security guards to permit local logging on to the system
 - Using physical access to perform logical attacks shows the complex interaction between levels, showing the need for a systematic model
- Gaining remote access shown within its own logical row marked 1b
- Misuses authorised access using own account to install backdoor
 - Immediate effect to retain access after leaving and avoids accountability
 - Similar act (not shown) at the physical level, would be to install a hidden ADSL modem or a wireless access point to retain physical access
- Access stage is not a complete incident, as security breaches do not usually interfere directly with organisational goals
 - Shown by the arrows not reaching the ultimate effects column
 - Instead passes down to the start of a subsequent stage using the acquired access
 - Has indirect effect of causing loss of system integrity and recovery costs

Access stage (3)

- Installed a logic bomb (persistent access) to delete the production software and all backups
 - Using compromised account of his colleague along with remote access
- Arrows cross the levels of the table showing informally the passage through a logical or physical boundary to gain lower-level access
- Table shows security breaches by distinguishing between
 - Unauthorized access that requires a traversal of the table from left to right first such as 1a to gain access via a compromised account
 - Authorized access marked 1b passes forward without requiring an initial access violation stage when setting up remote access with own account
- Table also demonstrates how the access gained in the first stage is used to gain further access
 - The installation of the logic bomb requires both the remote access and account compromise at point ♥
- Shows how our model captures the attack tree notion, although it has wider application
 - The two types of access needed to install the logic bomb would be an AND-node in an attack tree
 - Alternate paths to same goal captured by OR-nodes

Use stage

- Installation of the logic bomb is the launch pad for the use stage
 - Detonation of the bomb to destroy the production software is shown in the lowest logical row marked 2
- Effect of a sabotage incident is often coincident with the use stage
 - Execution of the logic bomb has the immediate effect of destroying the production software, and so consequentially, stage 2 is also marked 3
- Effect stage includes the subsequent use of the targeted resources by the perpetrator and escape
 - No escape needed as incident executed by the logic bomb, and the malware could self-destruct afterwards leaving no trace
- Resulted in the lost production of computers
 - Exacerbated by failure to provide independent protection for backups
 - Archer can destroy, steal, overwrite and otherwise interfere with backups
- Diagram shows conceptually how to provide defence-in-depth
 - Shows where independent controls can be placed to obstruct the use and effect stages after unauthorized access has been achieved
 - Temporal attack surfaces as opposed to spatial in structural diagrams

Alternative paths

- Clearly, there are many other paths that Archer could have followed, which should be investigated
 - Discovery is aided by our diagrammatic representation
 - However, needs greater clarity and automation
- Alternative paths model OR-nodes in attack trees
- Incidents can be analysed by locations and powers they can reach
 - Move within a connected conceptual space in our model
- Physical access could be obtained by tricking a security guard
 - Shown by a backwards arrow from gaining unauthorized access in the top row going to the beginning of the physical row analogous to ♦
 - Could then be used to steal or damage physical resources along a path in the bottom row
- Unconnected arrow marked ♠ in the bottom row, where physical destruction or removal of the software container has a logical effect
 - Again, all logical resources fundamentally dependent on physical embodiment
 - Similarly, authorised physical access would be shown by a forward arrow from Archer to the bottom row

Effect stage

- Effect stage includes the subsequent use of the targeted resources by the perpetrator and their escape and cover-up
 - No escape needed here, but cover-up aided by self-destructing malware
- Impact at lower levels from damage to resources and services is eventually transformed into organizational issues at the social level
 - Access stages do not cause an ultimate effect, except to cause disruption to find and repair the exploited weaknesses
 - Means to an end only shown by not reaching the ultimate effect column
- Sabotage has the immediate effect of compromising availability and integrity of the targeted resources (means) usually at lower layers
 - Employees are well positioned to target critical system weaknesses like essential components that are difficult to repair or replace
- Ultimate effect of damaging the organisation's ability to carry out its normal business activities (ends)
 - Shown as an arrow moving from a lower level to the social level in the last column demonstrating the ultimate effect

Ultimate impact

- Organisational context is crucial in security
 - Need to understand effects of security breaches
 - Argue that claims of component security like SSL are not meaningful
- Software controlling the production of computers was damaged (means) at the logical level
- Ultimate effect of damaging the organization's ability to produce and sell computers (ends)
- Caused financial damage and had knock-on effects on reputation and share price (ultimate ends) that met Archer's motivation (why)
 - Diagram should be consistent at social level and compared to lower levels, which can be modelled by logic
- Can interfere with the recovery mechanisms, which can be included in the incident table as additional steps that disrupt the defence
 - Archer achieved his ultimate objectives, because of the single point of failure of the production software that was centralized in one location
 - Destroyed backups that were all logically accessible by the logic bomb
 - Stopped effective recovery causing a significant outage rather than straightforward and speedy restoration from backups

Further issues

- Table shows the destruction of backup data marked •
- Destroying backups has little damaging impact alone (latent effect)
 - Effect is shown by a special arrow to the arrow between the logical and social levels strengthening the damage to the organization
 - Latent weakness causes damage only in conjunction with the loss of the primary system by intensifying its effects
- Must also consider the ultimate effect on the perpetrator, as he does not want to be held accountable (not always true)
 - Need to clearly distinguish between the goals of each side
 - Archer uses unauthorised and remote access and a logic bomb to deflect responsibility
 - Another method of avoiding detection is to turn off logging prior to the incident, or removing traces subsequently by deleting the log files
 - Shown as paths off the main track reachable within the incident space
- Physical attacks in particular are often overlooked and very hard to stop against determined insiders
 - Upward arrow marked ♠ shows the reliance of software and other logical resources on their physical storage containers

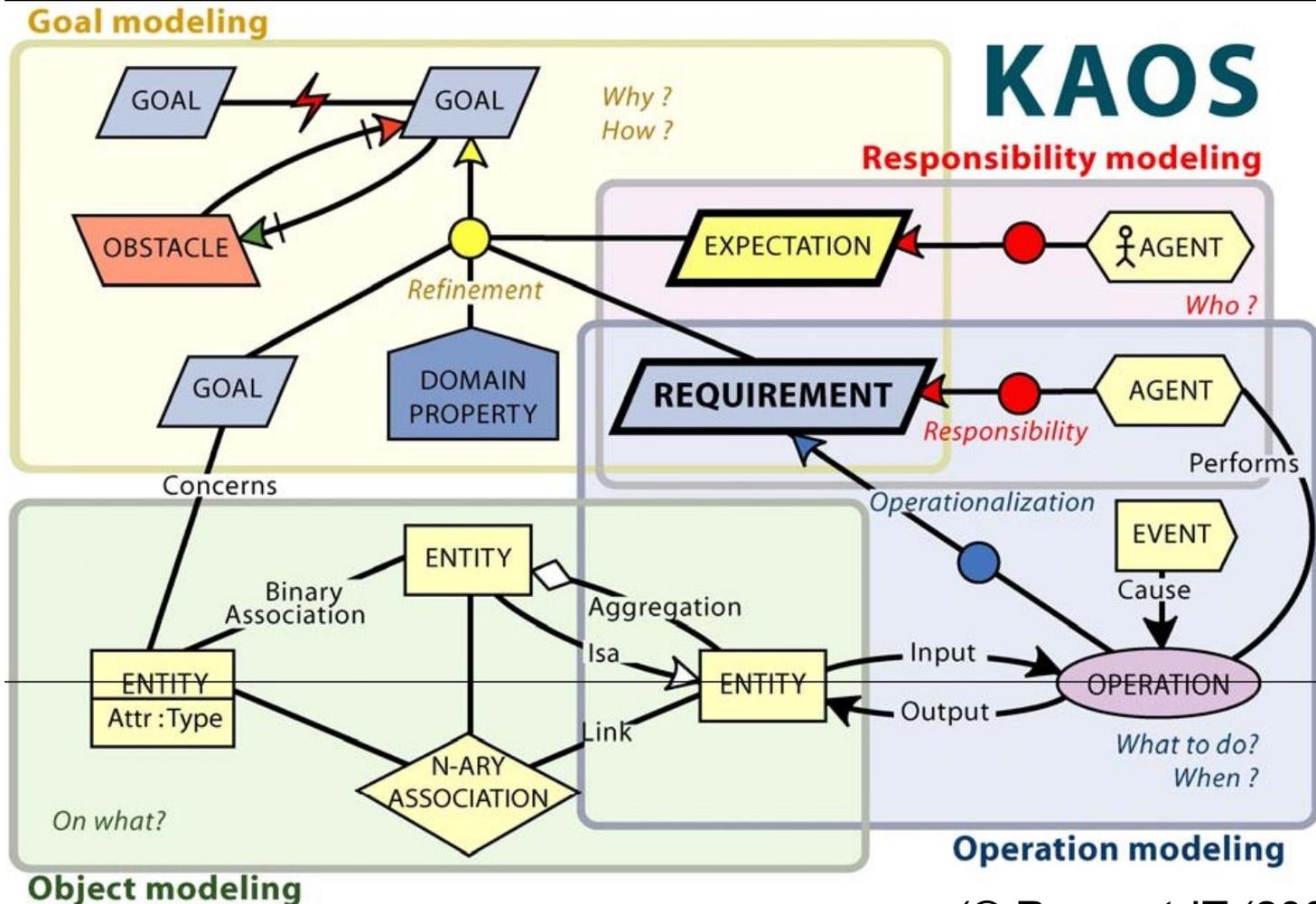
Table benefits

- Complete incident can be visualised in one table
- Incident achieves goals by moving locations, gaining powers or using resources modelled as movements in the table
- Can model complex attacks as it has a logical hierarchy with multiple stages
 - Incident divided into stages and stages link together by cause and effect
- Causes are on left of the table and effects on the right
 - States of entities and their environment (important for forensics)
- Activities (processes) are between the endpoint states
- Distinguish between control and real effects
 - Control effects are latent, but allow a real boundary to be breached
- Considers social level aspects such as motivation and goals
- Considers physical level as all incident aspects occur physically
- Each level has its own conception of space and time
- Each level has its own conception of action and state
 - Processes and data may be understood differently at every level

Insider threat from sabotage

Perpetrator (who) and Motivation (why)	Stage agent (who)	Reason (why)	Method (with what)	Action (how)	Target (to what)	Immediate Effect (what)	Ultimate effect (what)
Disgruntled former employee, Ian Archer. Motivation is psychological satisfaction from revenge for perceived mistreatment	Archer. 1a The targets unwittingly help to give unauthorized access to Archer 1b	To gain system access	Social engineering ♣ ∞ 1a alternative paths using email or phone (arrow extends to the physical row)	Persuade or trick target to act incorrectly by giving access, setting up accounts, or giving out passwords	Security guards, system administrators, colleagues	Unauthorized physical, or logical access (via a compromised account)	Inability to produce computers → Failure to satisfy contracts → Financial losses → Reduced reputation, lost customers, lowered share price
Social	1a						
Logical	1b Archer using his account	To gain hidden logical access after termination and avoid accountability	Misuse authorized authority using own account to issue commands	Logically authorized (but prohibited by policy at the social level) commands to set up a backdoor	Network access to system	Gain unauthorized remote logical access after termination	
Logical Remote	1c Archer using a compromised account and remote access	To install malware to maintain indirect system access and avoid accountability	Use of compromised account and remote access to issue commands	Unauthorized commands to install logic bomb	Operating system of computer holding production software	Installation of logic bomb <i>Backup loss</i>	
Logical Local	2/3 Logic bomb	To cause immediate damage to software → Ultimate goal to damage computer production	Privileged software misuses host system	Issue damaging commands to destroy files and software → May also cover tracks by deleting log files	Software on production control system and backups → Production processes Log files ♠	Unavailable production software → Lost computer production	
Physical	Archer <i>Physical attack goes from left to right cell to cell same as logical attack</i>	Render critical assets unavailable → Stop production line	Misuse allowed access to physically interfere with equipment and resources before leaving, illegitimate access after termination	Physical damage and destruction, theft, encryption	Software, backups, production computers and other essential equipment Logging devices	Damaged or unavailable systems and resources → Lost computer production	

KAOS Methodology

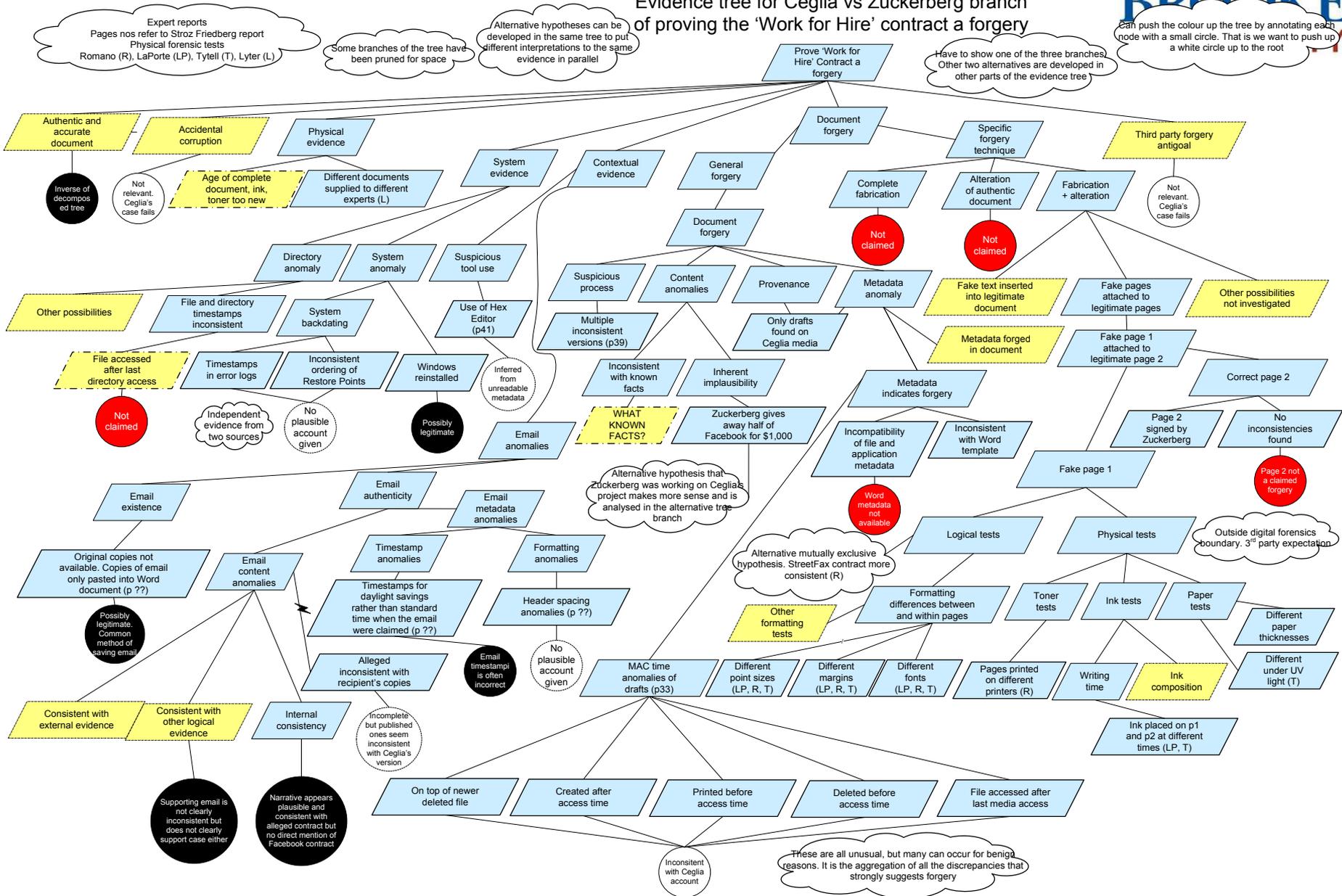


(© Respect-IT (2007))

Ceglia v Zuckerberg and Facebook

- Paul Ceglia produced what he said was a copy of the contract he and Mark Zuckerberg had signed
- Covers two projects on which the two were working together
- Ceglia project called "StreetFax" and a Zuckerberg project called "the face book."
- Strong evidence for a contract about StreetFax, but little for a contract involving Facebook
- However, Ceglia and Zuckerberg clearly did discuss the idea in detail, and the email give some support to Ceglia's version of events
- Ceglia paid Zuckerberg \$1,000 for work on StreetFax and alleges he paid \$1,000 to fund Zuckerberg's "face book" project
- According to Ceglia, the agreement says that Ceglia will get 50% of the "face book" project in exchange for funding initial development
- Four tree branches, each of which we decompose into demonstrable statements (~ requirements) to disprove Ceglia's case
- Next slide has one branch used to prove Ceglia's contract a forgery

Evidence tree for Ceglia vs Zuckerberg branch of proving the 'Work for Hire' contract a forgery



Expert reports
Pages nos refer to Stroz Friedberg report
Physical forensic tests
Romano (R), LaPorte (LP), Tytell (T), Lyter (L)

Some branches of the tree have been pruned for space

Alternative hypotheses can be developed in the same tree to put different interpretations to the same evidence in parallel

Have to show one of the three branches. Other two alternatives are developed in other parts of the evidence tree

Can push the colour up the tree by annotating each node with a small circle. That is we want to push up a white circle up to the root

WHAT KNOWN FACTS?

Alternative hypothesis that Zuckerberg was working on Ceglia's project makes more sense and is analysed in the alternative tree branch

Alternative mutually exclusive hypothesis. StreetFax contract more consistent (R)

These are all unusual, but many can occur for benign reasons. It is the aggregation of all the discrepancies that strongly suggests forgery

Conclusions

- Our incident process integrates with the structural model to create a comprehensive framework
 - As shown with the insider threat example
- Layered model considers people and physical world as well
 - Inspired by OSI model, but also includes processes and state
- Organizes incidents into stages of access, use and outcome
 - Allows comprehensive analysis from initiation to ultimate goal
- Use incident questions inspired by Zachman's framework
 - Can pose questions about the entire incident and each stage
- Analysis can be structured using KAOS and incorporated into architectural model
 - Using goal or attack (antigoal) trees

Issues

- Too complex to model new or predict incidents
 - Use KAOS to decompose incidents systematically
- Formalisation
 - Diagrammatic representation
 - Can model by UML diagrams at each level
 - Use formal graphical method called bigraphs (spygraphs)
 - Graphical model is the data, and rewriting rules is the program
 - Logical representation (for syntax)
 - Use Event calculus to model location and time
 - Model theory (for semantics)
 - Ontology based on Sowa's upper ontology