

Software Security Principles and Guidelines

Samuel T. Redwine, Jr.
James Madison University

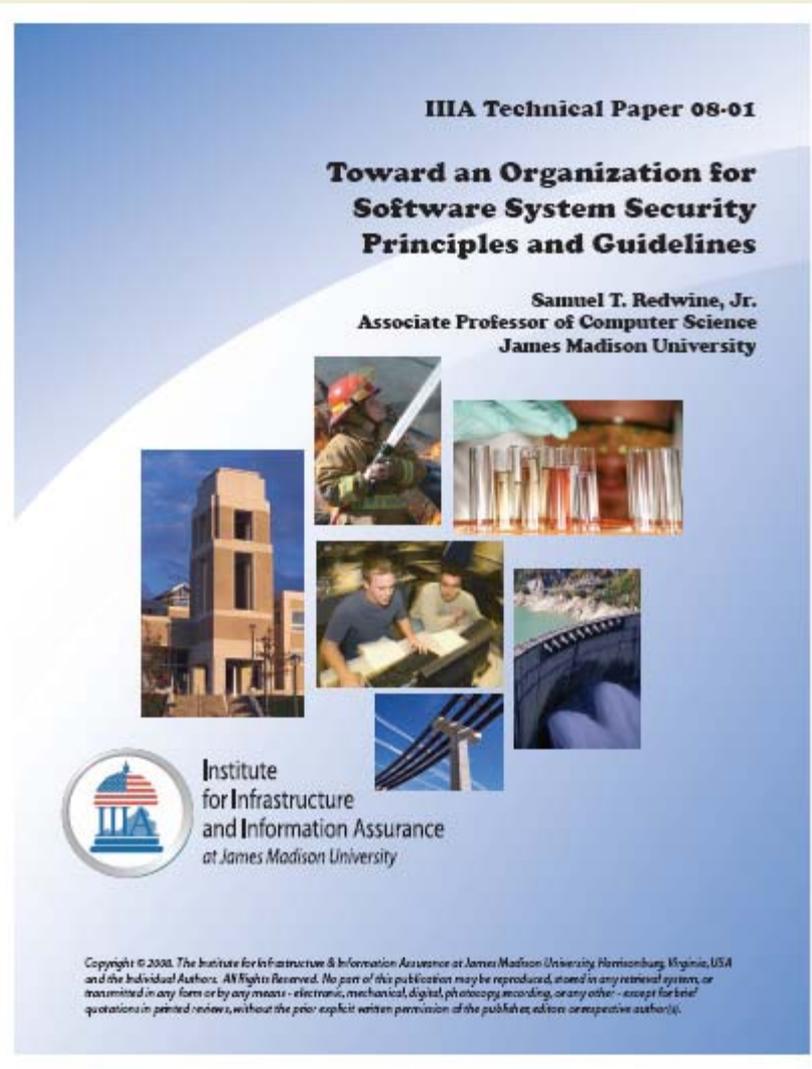


Agenda

- Introduction (5 minutes)
- High-level Structure (5 minutes)
- Some Themes (8 minutes)
- Quick Overview (15 minutes)
- Selected Areas (15 minutes)
- Conclusion (2 minutes)
- Questions and Answers (10 minutes)

Introduction

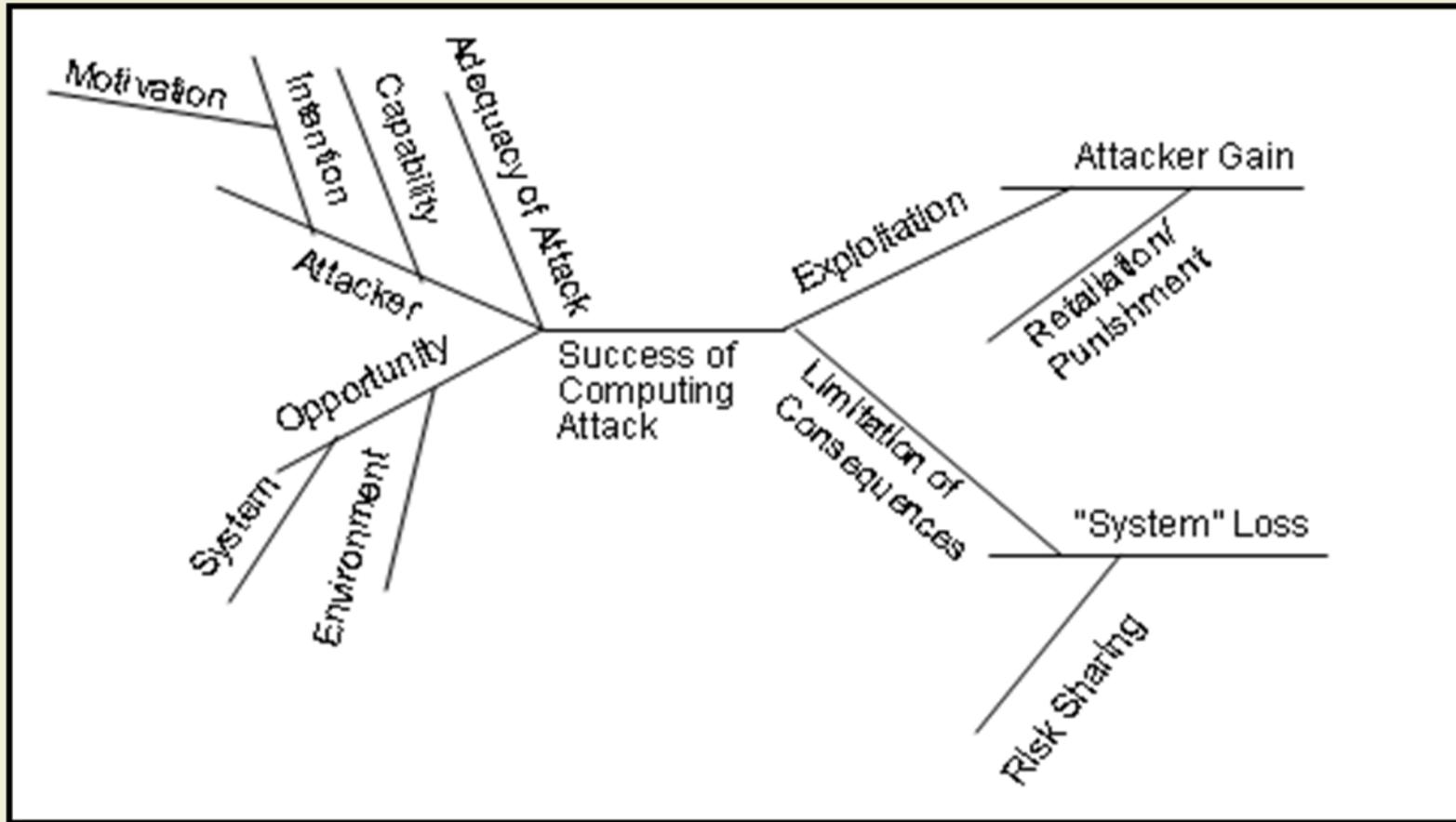
Within time available can only give an overview.



Principles and Guidelines

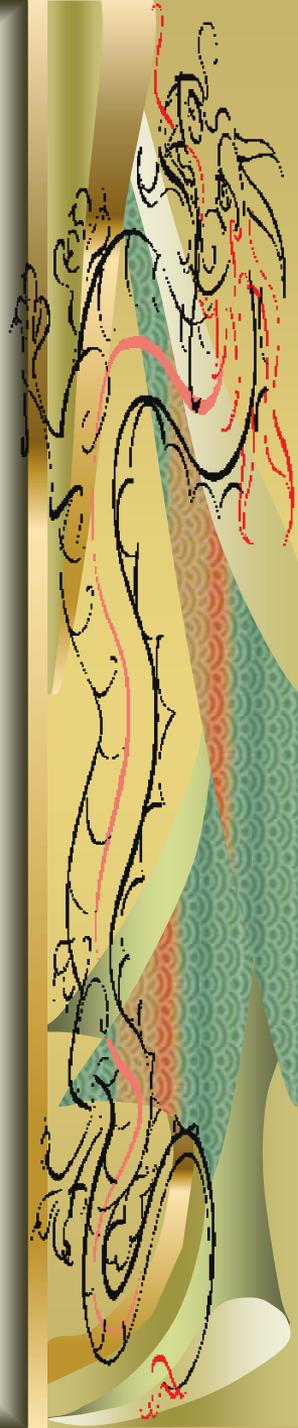
- Vary from
 - Idealized to practical
 - Absolute commands to conditional suggestions
 - Universal to particular
 - Entirely accurate to approximate (e.g. incomplete)
- Famous ones from Saltzer and Schroeder's seminal 1975 article appear at appropriate places and levels in report
- **Today, discuss only high-level items**
 - **And not argue about what is a principle and what a guideline**

Background: Simplified Security "Cause and Effect" Diagram



Top Two Levels of Structure

Hierarchical Structure with
Some Duplication at Lower
Levels



Top Level of Structure

- **The Adverse**

- **The System**

- **The Environment**

- Could be thought of as roughly Attackers, Defenders, and Arena

Structure

■ Top Level

- **The Adverse**

- **The System**

- **The Environment**

■ Second Level under each of these three

- Related entities and phenomena

- Benefits

- Losses

- Uncertainties

Often about understanding and limiting these

Top Two Levels

1. THE ADVERSE

1.1. Limit, Reduce, or Manage Violators
(Including attempts to violate)

1.2. Limit, Reduce, or Manage Benefits to Violators or Attackers

1.3. Increase Attacker Losses

1.4. Increase Attacker Uncertainty

2. THE SYSTEM

2.1. Limit, Reduce, or Manage Violations

2.2. Improve Benefits or Avoid Adverse Effects on System Benefits

2.3. Limit, Reduce, or Manage Security-related Costs

2.4. Limit, Reduce, or Manage Security-related Uncertainties

3. THE ENVIRONMENT

3.1. Nature of Environment

3.2. Benefits to and from Environment

3.3. Limit, Reduce, or Manage Environment-Related Losses

3.4. Limit, Reduce, or Manage Environment-Related Uncertainties

Some Underlying Themes

Provide Background and
Aid in Understanding



First Theme: Limit, Reduce, or Manage

- Limit, reduce, or manage **undesirable** entities, conditions, or events
 - Opportunities (offered) for
 - Preconditions for
 - Motivation, intention, and capability to try and/or succeed
 - Attempts to
 - Occurrences of
 - Consequences from
 - Lacks
 - Lack of knowledge, understanding, and detection
 - Lack of accountability
 - Lack of learning and improvement

Second Theme: Maliciousness

- Violations and adverse consequences can result from entities or acts that are either
 - Malicious
 - Non-malicious
- Thus, violators can be either of these
- The existence of maliciousness does not remove non-malicious difficulties

More Themes

■ Software is in danger all its life

- Throughout Life cycle

■ Assurance

- Need grounds for confidence

- Assurance case: (Security-related) Claim and its justification including arguments and evidence
- Build the right thing, build it right, show it is right

- Know degree of accomplishment and uncertainty regarding it

- Claim could be regarding behavior, conditions, violations, and/or consequences

■ Software system security is a conflict: a substantial amount known about conflicts

Slogan Expanded

■ Do the right thing

- System (preferably as specified)
 - Predicted beneficial enough given predicted lifecycle costs
 - Not too (downside) dangerous (given degree of risk aversion)
 - Enough (upside) opportunities (given degree of ambitiousness)
 - Compliant with laws, ethics, etc.

■ Do it right

- What is intended is what is created
- All activities and artifacts over entire lifecycle
 - Well-engineered, well-managed, well-supported, well-marketed, etc.

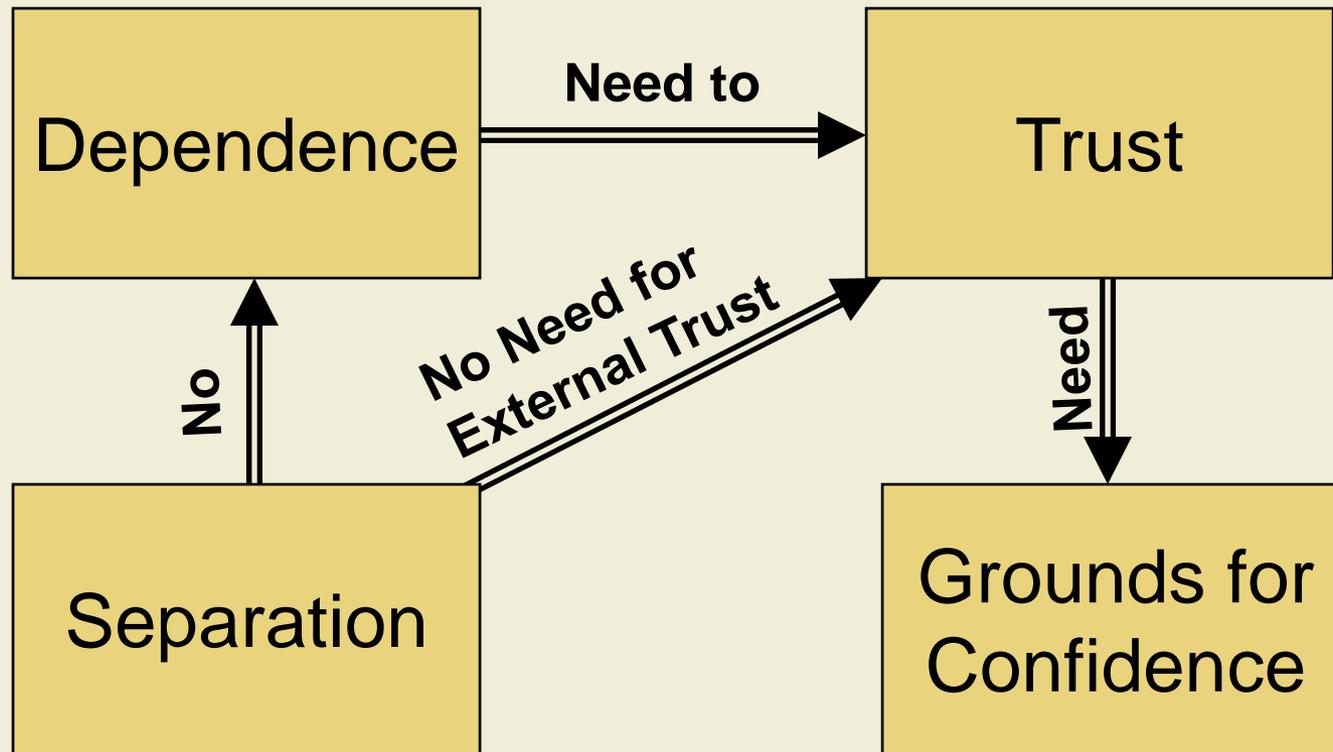
■ Show it is right

- So sure enough about 1 and 2 that
 - Stakeholders can make adequate initial and on-going decisions
 - Uncertainties do not result in unacceptable or intolerable uncertainties about occurrences or sizes of potential consequences

Four Related Topics

Two
Entities

Implications



Assurance Case

Three Related Topics (cont'd)

■ Dependence

- Limit dependence
 - Constrain (e.g. criticality)
 - Require collusion
 - Or multiple mistakes or combination
 - Nothing unnecessary
 - Localize
 - Number, size, complexity
- Minimize security elements
 - Exclude non-security functionality
- Avoid dependence on environment
- Dependence means exposure

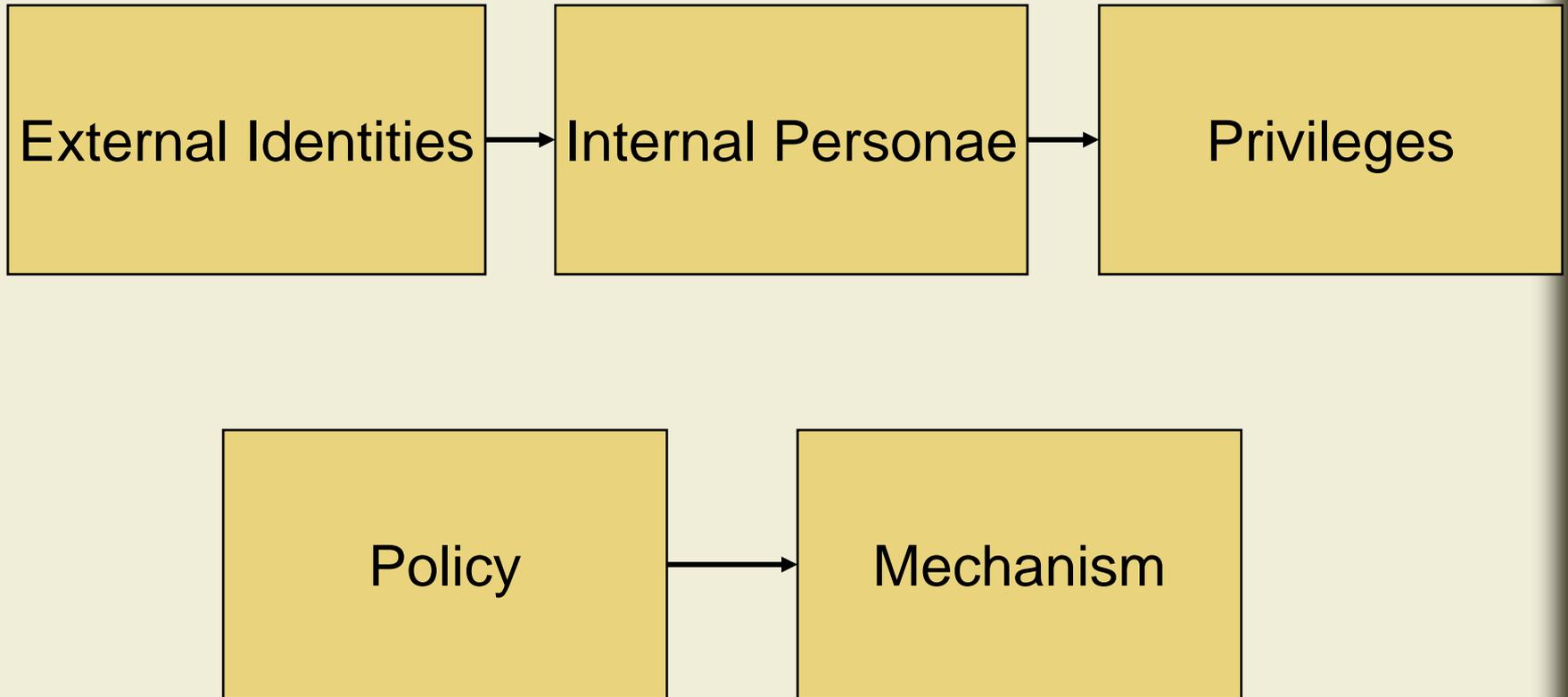
■ Trust

- Minimize what must be trusted
- Trust only adequately trustworthy
- Limit the trust extended
- Trust does not scale well

■ Separation

- Separation protects
- Separate from dangers
- Ensure all means of interaction are known
 - No bypassing
- Limit paths for interaction
 - Minimize sharing

Separation of Concerns



Conclusion

Several themes

-  Exist within set of principles and guidelines and
-  Underlie lower levels of their organization

One Pass through the Structure



1. THE ADVERSE

1.1. Limit, Reduce, or Manage Violators

- 1.1.1. *Adversaries are Intelligent and Malicious*
- 1.1.2. *Limit, Reduce, or Manage Set of Violators*
- 1.1.3. *Limit, Reduce, or Manage Attempted Violations*
- 1.1.4. *Think like an Attacker*

1.2. Limit, Reduce, or Manage Benefits to Violators or Attackers

- 1.2.1. *Unequal Attacker Benefits and Defender Losses*
- 1.2.2. *Limit, Reduce, or Manage Violators' Ability to Exploit Success for Gain*

1.3. Increase Attacker Losses

- 1.3.1. *Limit, Reduce, or Manage Violators' Ease in Taking Steps towards Fruitful Violation*
- 1.3.2. *Increase Losses and Likely Penalties for Preparation*
- 1.3.3. *Increase Expense of Attacking*
- 1.3.4. *Increase Attacker Losses and Likely Penalties*

1.4. Increase Attacker Uncertainty

- 1.4.1. *Conceal Information Useful to Attacker*
- 1.4.2. *Exploit Deception*

2. THE SYSTEM

2.1. Limit, Reduce, or Manage Violations

- 2.1.1. Specify Security Requirements
- 2.1.2. Limit, Reduce, or Manage Opportunities for Violations
- 2.1.3. Limit Reduce, or Manage Actual Violations
- 2.1.4. Limit, Reduce, or Manage Lack of Accountability

2.2. Improve Benefits or Avoid Adverse Effects on System Benefits

- 2.2.1. Access Fulfills Needs and Facilitates User
- 2.2.2. Encourage and Ease Use of Security Aspects
- 2.2.3. Articulate the Desired Characteristics and Tradeoff among Them
- 2.2.4. Efficient Security
- 2.2.5. Provide Added Benefits
- 2.2.6. Learn, Adapt, and Improve

2.3 Costs and 2.4 Uncertainty are on next chart

Example Extract: Protect Valuables Everywhere Always

- Continuous Protection of Assets
- Protect It Everyplace It Goes
 - *End-to-end Protection*
 - *Protect All Media*
- Protect (All) Copies
- Protect all Forms or Guises

Similarly but more idealistically:

- Eliminate (All) Hazards
- Protect against All Threats
 - *Guard All Approaches*
 - *Guard Adequately*

2. THE SYSTEM (cont'd)

2.3. Limit, Reduce, or Manage Security-related Costs

2.3.1. *Limit, Reduce, or Manage Security-Related Adverse Consequences*

2.3.2. *Limit, Reduce, or Manage Security-Related Expenses across the Lifecycle*

2.4. Limit, Reduce, or Manage Security-related Uncertainties

2.4.1. *Identify Uncertainties*

2.4.2. *Limit, Reduce, or Manage Security-Related Unknowns*

2.4.3. *Limit, Reduce, or Manage Security-Related Assumptions*

2.4.4. *Limit, Reduce, or Manage Lack of Integrity or Validity*

2.4.5. *Limit, Reduce, or Manage Lack of Reliability or Availability of Security-related Resources*

2.4.6. *Predictability – Limit, Reduce, or Manage Unpredictability of System Behavior*

2.4.7. *Informed Consent*

2.4.8. *Limit, Reduce, or Manage Consequences or Risks related to Uncertainty*

2.4.9. *Increase Assurance regarding Product*

3. THE ENVIRONMENT

3.1. **Nature of Environment**

3.1.1. *Security is a System, Organizational, and Societal Problem*

3.1.2. *The Conflict Extends beyond Computing*

3.1.3. *New Technologies Have Security Problems*

3.2. **Benefits to and from Environment**

3.2.1. *Utilize Security Mechanisms Existing in Environment to Enhance One's Security*

3.2.2. *Create, Learn, and Adapt and Improve Organizational Policy*

3.2.3. *Learn from Environment*

3.2.4. *Help, but do not Help Attackers*

3.3. **Limit, Reduce, or Manage Environment-Related Losses**

3.3.1. *Do Not Cause Security Problems for Systems in the Environment*

3.3.2. *Do Not Thwart Security Mechanisms in Environment*

3.3.3. *Avoid Dependence*

3.3.4. *Presume Environment is Dangerous*

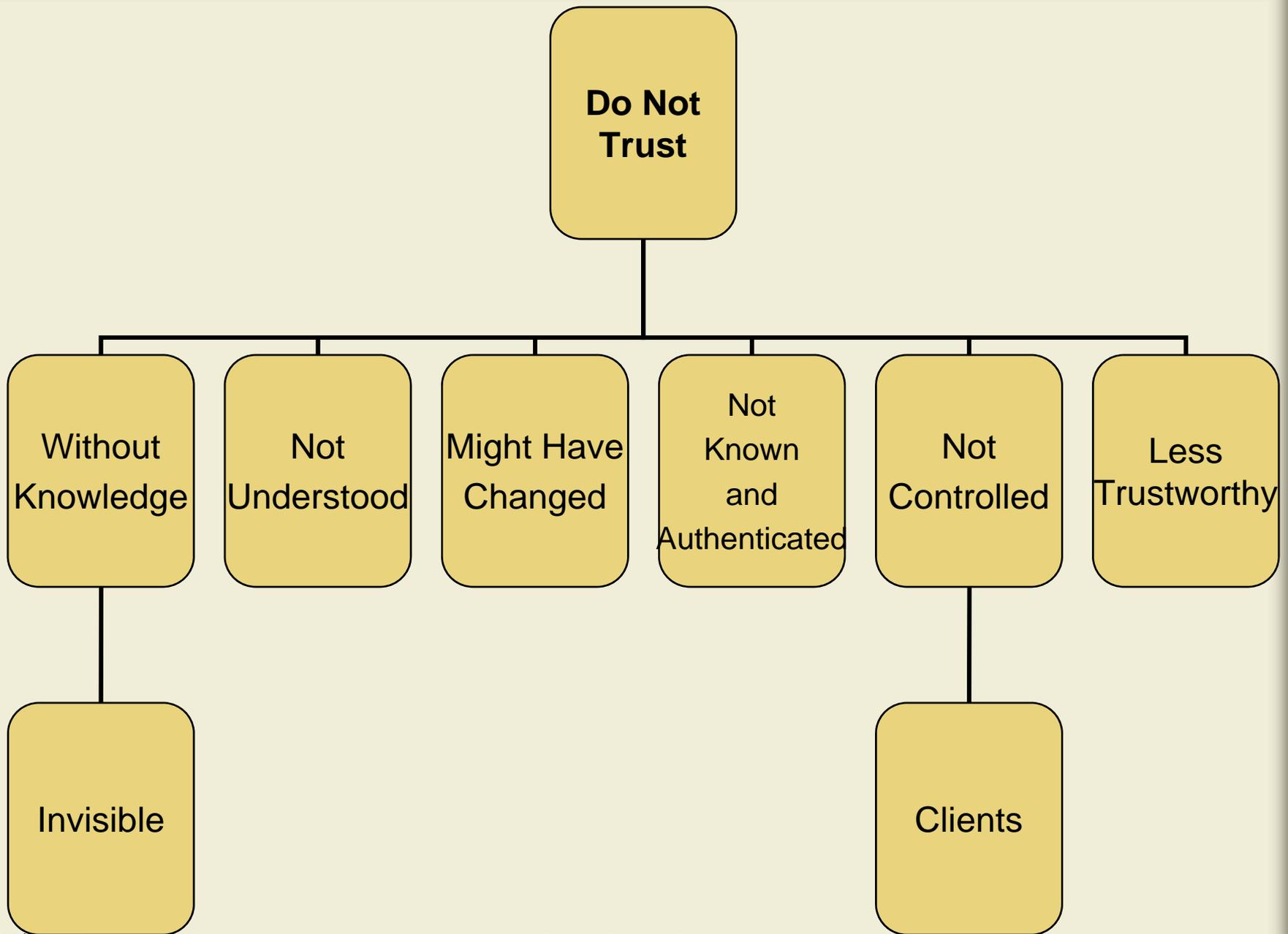
3.4. **Limit, Reduce, or Manage Environment-Related Uncertainties**

3.4.1. *Know One's Environment*

3.4.2. *Limit, Reduce, or Manage Trust*

3.4.3. *Ensure Adequate Assurance for Dependences*

3.4.4. *Third-Parties are Sources of Uncertainty*



More Detail

Sections on The System's
2.3 Losses
2.4 Uncertainties



2.3. Limit, Reduce, or Manage Security-related Costs

2.3.1 *Limit, Reduce, or Manage Security-Related Adverse Consequences*

- All Actions have Consequences
- Losses can take Many Forms
- Values of a Consequence Vary among Stakeholders
- Predict Consequences
- Limit, Reduce, or Manage Post-Violation Consequences
- Tolerate Security Violations
- Recover
- Support Forensics and Incident Investigations
- Allocation of Defenses according to Consequences
- Software that is Malicious or Susceptible to Subversion is as Dangerous as Humans who are Malicious or Susceptible to Subversion

2.3.2. *Limit, Reduce, or Manage Security-Related Expenses across the Lifecycle (next chart)*

2.3. Limit, Reduce, or Manage Security-related Costs

2.3.2. Limit, Reduce, or Manage Security-Related Expenses across the Lifecycle

- Limit, Reduce, or Manage Security-Related Developmental and Operational Expenses
- Cannot Retrofit Security
- Ease Downstream Security-related Activities
- Reuse only Adequately Specified and Assured Components

2.4. Limit, Reduce, or Manage Security-related Uncertainties

2.4.1. Identify Uncertainties

- Identify Sources of Uncertainty
- Identify Individual Uncertainties
- Identify Relationships among Uncertainties

2.4.2. Limit, Reduce, or Manage Security-Related Unknowns

2.4.3. Limit, Reduce, or Manage Security-Related Assumptions

- Reasoned Assumptions
- Avoid Critical Assumptions

2.4.4. Limit, Reduce, or Manage Lack of Integrity or Validity

- Representation of Reality is Not Reality
- Possible Lack of Integrity is a Source of Uncertainty
- Limit, Reduce, or Manage Lack of Integrity or Validity of Security-related Resources

2.4.5. Limit, Reduce, or Manage Lack of Reliability or Availability of Security-related Resources

2.4. Limit, Reduce, or Manage Security-related Uncertainties

2.4.6. Predictability – Limit, Reduce, or Manage Unpredictability of System Behavior

- Use repeatable engineering process, means, and environment to produce predictably behaving product
- Ensure Engineering Artifacts Exist that Show How System Meets Assured Requirements
- Verifiability

2.4.7. Informed Consent

2.4.8. Limit, Reduce, or Manage Consequences or Risks related to Uncertainty

- Continuous Risk Management
- Risk Sharing

2.4. Limit, Reduce, or Manage Security-related Uncertainties

2.4.9. Increase Assurance regarding Product

- System Assurability
- Reduce Danger from other Software or Systems
- Limit or Reduce Complexity
- Predictable Change
- Change Slowly
- Assure Security of Product
- Use Production Process and Means that Ease and Increase Assurance

Conclusion

- Coherence and comprehensiveness
- Top-down organization of principles and guidelines possible
- Aid to organizing one's understanding and in curriculum development

[www.jmu.edu/iiia/webdocs/Reports/SwA Principles Organization-sm.pdf](http://www.jmu.edu/iiia/webdocs/Reports/SwA_Principles_Organization-sm.pdf)

I welcome comments. Send to redwinst@jmu.edu.

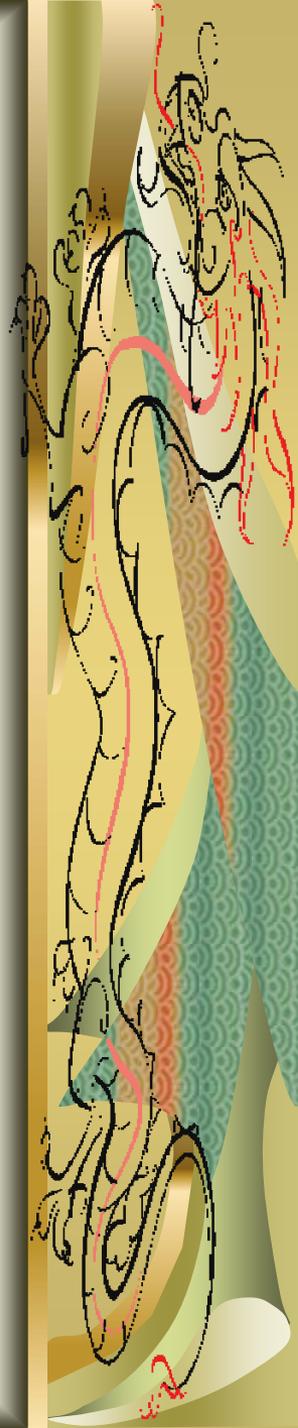
Questions and Discussion

<i>Analogize</i>	<i>Conceptualize</i>	<i>Hypothesize</i>	<i>Ponder</i>
<i>Analyze</i>	<i>Conjecture</i>	<i>Infer</i>	<i>Propose</i>
<i>Apply</i>	<i>Discover</i>	<i>Imagine</i>	<i>Question</i>
<i>Agree</i>	<i>Discriminate</i>	<i>Integrate</i>	<i>Reason</i>
<i>Argue</i>	<i>Estimate</i>	<i>Invent</i>	<i>Recount</i>
<i>Assert</i>	<i>Evidence</i>	<i>Judge</i>	<i>Specialize</i>
<i>Calculate</i>	<i>Examine</i>	<i>Link</i>	<i>Solve</i>
<i>Caution</i>	<i>Explain</i>	<i>Measure</i>	<i>Suppose</i>
<i>Claim</i>	<i>Extrapolate</i>	<i>Observe</i>	<i>Theorize</i>
<i>Clarify</i>	<i>Foresee</i>	<i>Opine</i>	<i>Validate</i>
<i>Conceive</i>	<i>Generalize</i>	<i>Organize</i>	<i>Verify</i>

Document Availability

- *Towards an Organization for Software System Security Principles and Guidelines* version 1.0, by Samuel T. Redwine, Jr., Institute for Infrastructure and Information Assurance, James Madison University, IIA Technical Paper 08-01. February 2008. Available at [www.jmu.edu/iiia/webdocs/Reports/SwA Principles Organization-sm.pdf](http://www.jmu.edu/iiia/webdocs/Reports/SwA_Principles_Organization-sm.pdf)
- *Software Assurance*, Samuel T. Redwine, Jr. (Editor), US DHS 2006. Available at
 - <https://buildsecurityin.us-cert.gov/daisy/bsi/940.html>
- May want to look next at
 - *High-Assurance Design*, Clifford Berg, Addison Wesley, 2006
- Also of possible interest is revision underway of ISO/IEC (and IEEE) 15026 System and Software Assurance

Additional Charts



Actions and Results

- Prepare
 - Develop Capability to
 - Develop Intention to
- Affect Others' Preparations
- Offer or Allow Opportunities
 - To Opposition or Environment
- Perceive Opportunity
- Do (e.g. Attack, Defend)
 - Attempt to
 - Actually Perform at Some Level of Proficiency
- Succeed (Partially or Wholly) or Fail
 - Potentially Yielding
 - Benefits, Losses, and Uncertainties
 - New Situation
- Affect Actual Follow-on Consequences
 - Benefits, Losses, and Uncertainties
 - Sustained (Enhanced) Capability
 - Continued (or Strengthen) Intention

Assurance Case

