Software Supply Chain Risk Management: From Products to Systems of Systems

Carol Woody, Ph.D.

Supply Chains

Supply chain: set of suppliers that contribute to the content of a product or system or that have opportunity to modify its content. (Comprehensive National Cybersecurity Initiative 11)

- Hardware product involves multiple deliveries of the same item (built to specification)
- Software product is typically a single item redistributed within an organization

Supply-Chain Risk

Hardware supply chains – decades of data collection

- Manufacturing and delivery disruptions
- Manufacturing quality
- Counterfeit hardware estimated at 10%

Software – little data for software supply chains

- Third-party tampering during development or delivery
- Malicious supplier
- Compromised by inadvertent introduction of exploitable design or coding errors

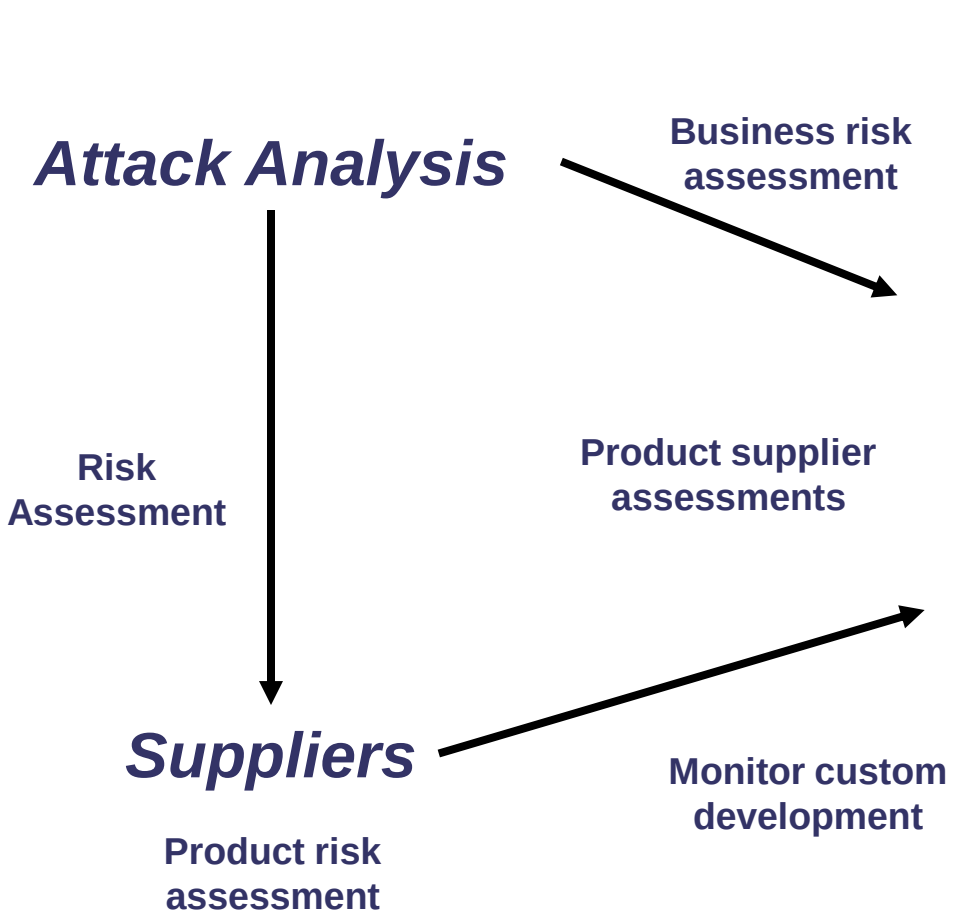
Software Supply Chain Security Risks

Attack analysis: factors that lead to successful attacks

Supplier: capability to limit product attributes that enable attacks

Acquirer: tradeoff decisions between desired use and acceptable business risk

Software Supply Chain Risk Management



Acquirers

Understanding what can be controlled

Supplier selection

Consequences

Possible tradeoffs

- accepted risks
- expense
- desired functionality
- network connectivity
- end-user access

Feasibility – limits of controls

Operations

Attack Analysis

Incentives & enablers

Value of data or service

Exploitable defects & features

Risk factors

Software dependencies

Network connectivity

End-user computing

Attacker intent

Consequences

Enablers: Software Errors

MITRE has documented software errors that have led to exploitable vulnerabilities: Common Weakness Enumeration (CWE)

CWE/SANS[1] Top 25 Most Dangerous Programming Errors published yearly by MITRE – 3/1/2010

Examples

Improper Input Validation	SQL Injection
Cross-site scripting	Use of Hard-coded Credentials
Download of Code Without Integrity Check	Improper Check for Unusual or Exceptional Conditions
Race Condition	Classic Buffer Overflow

1. http://cwe.mitre.org/top25/
SANS (SysAdmin, Audit, Network, Security) Institute

Veracode: State of Software Security

58% of all applications did not achieve an acceptable security score upon first submission Fall 2010

Measured Against CWE/SANS Top-25 Errors

Software Source	Acceptable
Outsourced	6%
Open Source	39%
Internally Developed	30%
Commercial	38%

Example: Stuxnet

Enabled the attacker to modify how the control system managed a physical system. General purpose control systems such as Siemens' execute user supplied software designed for the specific application.

Strategy:

To avoid detection, do not use corporate networks to directly modify the control system software

Use Internet access and defects in Windows or in application software to compromise computing resources belonging to trusted administrators – hundred of thousands of computers were actually compromised. – **Defects are an enabler, and network connectivity is a risk factor.**

Use computing resources such as the USB drives used by system administrators to transfer malware to the control systems **Use of end-user computing resources is a risk factor.**

Use control system extensibility to install control software that would adversely change the behavior of existing control functions. **Product feature is an enabler. No auditing or notification of control code changes are design faults.**

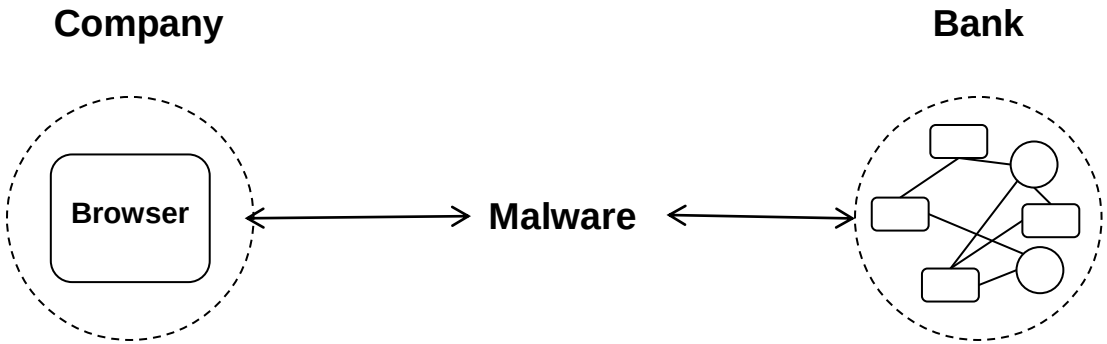
Example: Bank Fraud

Organizations with limited IT support – e.g. school districts

Organization's computer used for bank transaction is compromised

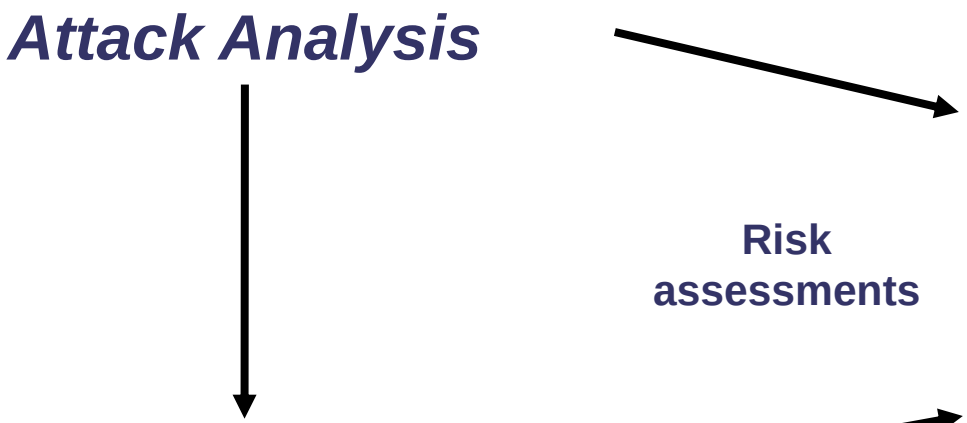
Malware deployed that that receives and can transforms web pages – man-in the middle

When user logs into financial system, a page is returned that informs the user that there will short delay (while malware submits transactions)



Frequent design fault: Financial systems assumed client has not been compromised. Confirmations for fraudulent transactions returned over compromised communications path and blocked by the malware.

Software Supply Chain Risk Management



Acquirers

Understanding what can be controlled

Supplier selection

Consequences

Possible tradeoffs

- accepted risks
- expense
- desired functionality
- network connectivity
- end-user access

Feasibility – limits of controls

Operations

Supplier: Attack Surface Analysis

Reduce Attack Surface

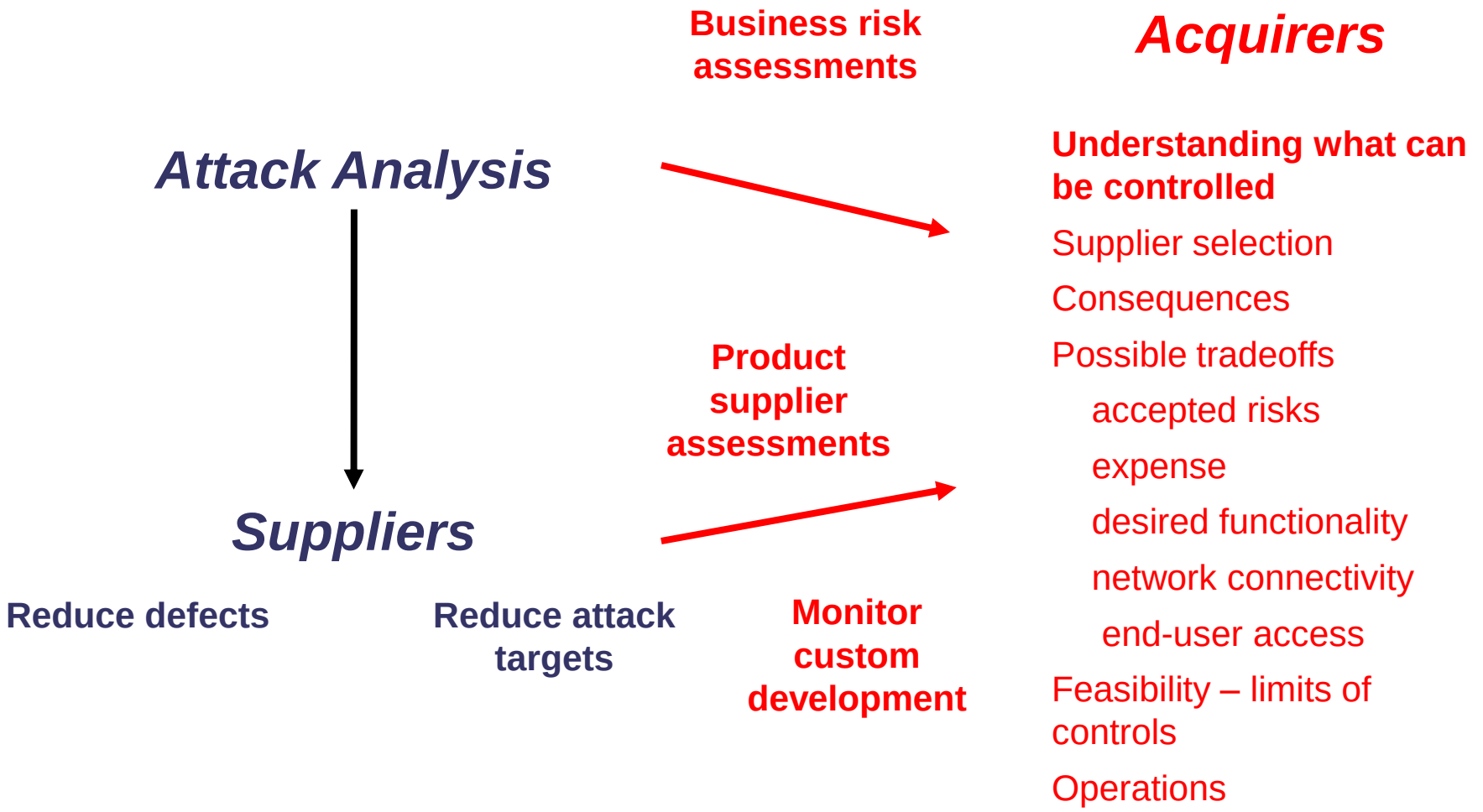
- Remove or change system features or re-architect the implementation to avoid attack enablers or unnecessary channels.
- Revise use of an emerging technology where there is limited knowledge of the potential exploits and mitigations
- Review requirements or implementation if existing mitigations are costly or do not provide the necessary assurance

Supplier: Risk Focused Development

Data flow analysis (threat modeling)

- Consider known weaknesses and attack patterns – e.g. mix of data and commands
- Document security assumptions and trust boundaries
- Consider deployed configuration and expected usage
- Analyze the interfaces to other components (inputs and outputs)
- Consider consequences
- Analyze possible mitigations
- Provide architecture and design guidance

Software Supply Chain Risk Management

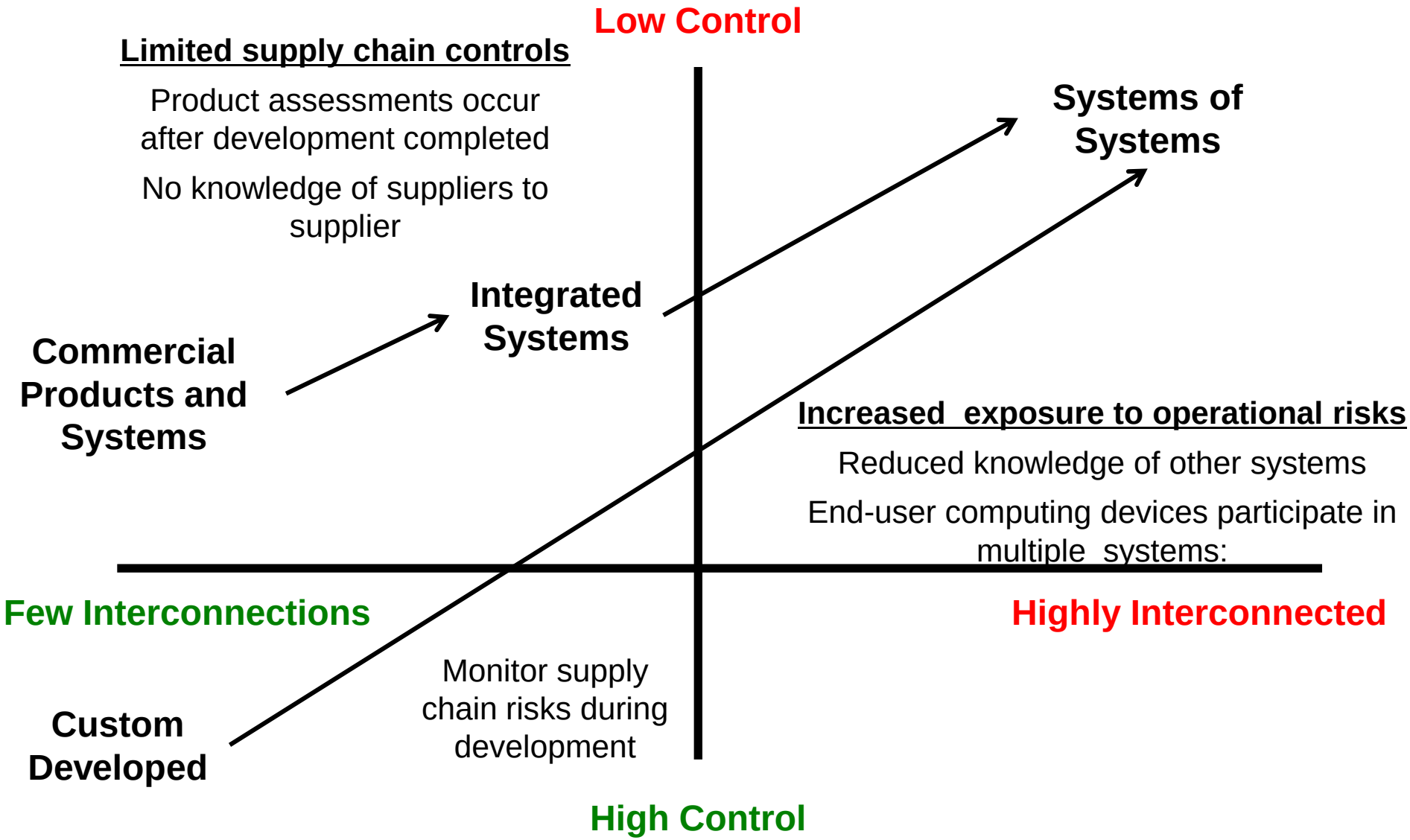


Acquirers

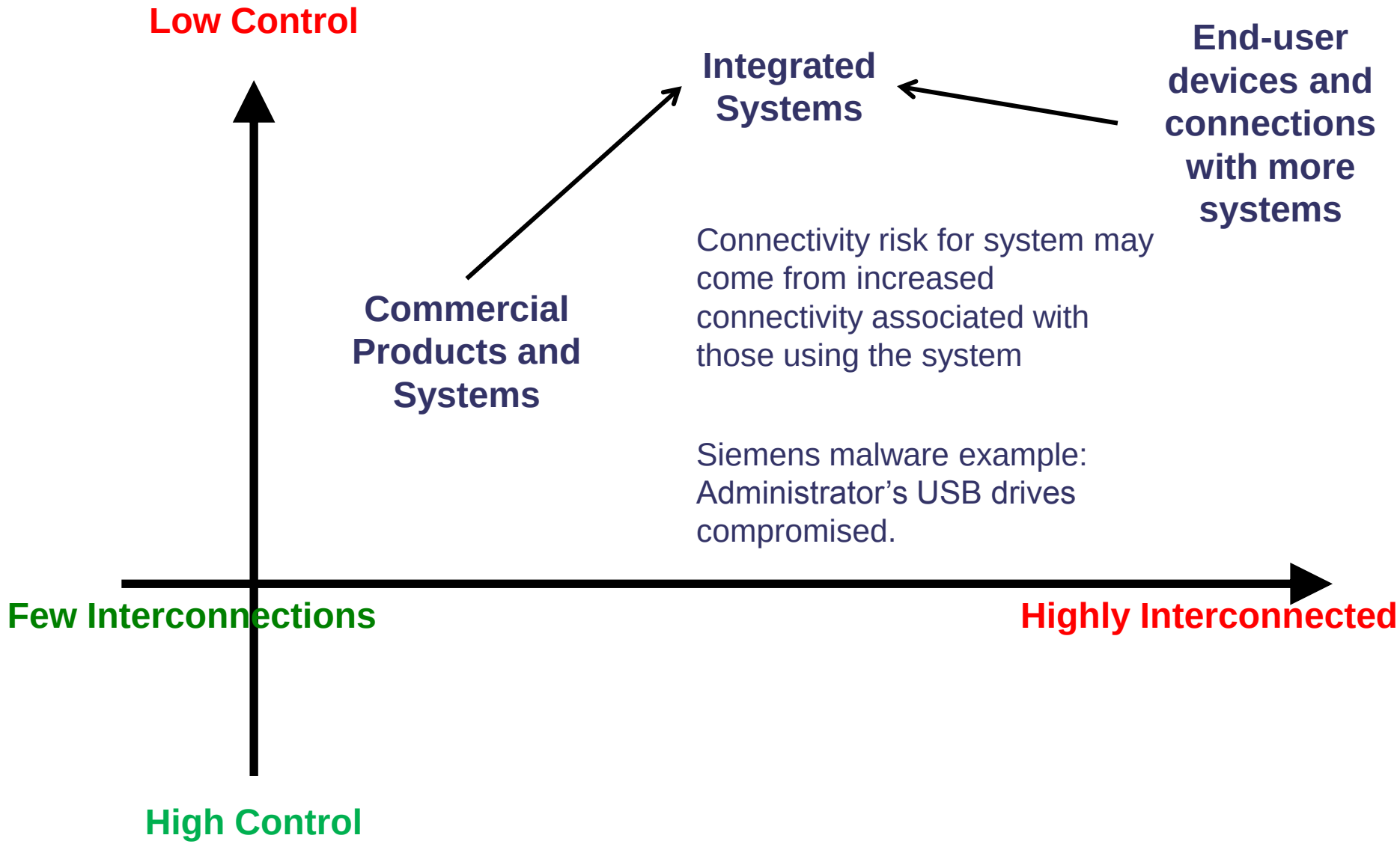
Acquisition control points depend on the kind of type of acquisition

- Specific products – end-user software
- Software products incorporated into a system
- System development and integration

Connectivity and Control[1]



Connectivity and Control[2]



Limitations of Supply Chain Risk Management

Limited visibility of supply chain

Uncertainty of product assurance

- Supplier relationship through acquisition (contracts)
- Attacks considered by supplier not known

Evolving nature of threats, usage, & product functionality

Impact of deployment and operations

Next Steps

Guidance

- SEI technical reports (2) available at sei.cmu.edu

Opportunities

- Supply Chain Risk Management Workshop
- On-site tutorials to tailor the guidance for specific organizational needs
- Pilot supply chain risk assessment for an acquisition

Sources

Evaluating and Mitigating Software Supply Chain Security Risks

- http://www.sei.cmu.edu/library/abstracts/reports/10tn016.cfm

Attack Surface

- Michael Howard, 2003, http://msdn.microsoft.com/en-us/library/ms972812.aspx

Threat Modeling

- Frank Swiderski, Window Snyder, *Threat Modeling*, 2004
- Michael Howard and Steve Lipner. *The Security Development Lifecycle,* 2006
- James McGovern, & Gunnar Peterson. “10 Quick, Dirty, and Cheap Things to Improve Enterprise Security.” Security & Privacy, IEEE, March-April 2010
- Building Security In Maturity Model (BSIMM) http://bsimm2.com/index.php
- John Stevens, “Threat Modeling— Perhaps It’s Time”, Security & Privacy, IEEE, May-June 2010

NO WARRANTY

THIS MATERIAL OF CARNEGIE MELLON UNIVERSITY AND ITS SOFTWARE ENGINEERING INSTITUTE IS FURNISHED ON AN “AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This Presentation may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at permission@sei.cmu.edu.

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.