The Large-Scale Remediation of Security Vulnerabilities

Presented by Bill Kowaleski, Hewlett-Packard Consulting and Integration

bill.kowaleski@hp.com

WHAT WE‘LL DO TODAY

- Define just what a vulnerability mitigation process really is and why it‘s important
- Discuss need for management buy-in
- Explain the case study we‘ll be referring to throughout the presentation
- Discuss how to decide just what vulnerabilities to mitigate and what *mitigate* really means
- Define different system types and their corresponding mitigation strategies
- Describe the functional roles needed
- Describe the processes needed
- Discuss timelines
- Discuss key lessons learned during case study

What is Vulnerability Mitigation?

- Part of a self-assessment program
- Consists of the following steps:
  - Scanning or other vulnerability discovery activities
  - Organization of resulting data
  - Assignment of mitigation activities
  - Installing patches, SP's, reconfiguring, etc.
  - Monitoring of progress
  - Technical support
  - Validation
- Easy on a small scale - A real challenge in a large, dispersed global organization

Why Do It?

- Increases in malicious damage to systems
- Code Red / Nimda - style worms can do tremendous damage quickly
- Protection of brand image
- Service level agreements
- May be only reasonable option today. Being strictly reactive is no longer feasible.
- Your job?

Management Support

- Resistance can be fierce and strong management support is essential
- Exceptions can be destructive and must be kept to a minimum
- Disconnection can lead to anger and appeals to upper management. They must be prepared in advance for this!
- Keeping management support is as important as getting it. Schedule regular sponsor meetings and keep sponsors fully-informed
- A sponsor committee with senior management from impacted areas works best

The Case Study

- Very Large, Global Organization
- In many different businesses
- Widely dispersed on 6 continents
- Over 5000 servers managed by over a dozen different groups
- Hundreds of Internet-accessible systems
- Entrepreneurial culture
- Reactive model for security incidents
- Hammered in late 2000 by hackers - many severe incidents
- I ran vulnerability mitigation program first 6 months

Vulnerability Discovery

- On a large scale, when you don’t control all the systems, scanning is only practical way
- Issues
  - Which scanner to use
  - Use more than one scanner?
  - Server discovery
  - What to do with all the results??
- High-risk systems may require additional tests
  - Penetration tests
  - Host-based security tools (expensive)

Vulnerability Classification

- Scanners have their own classification scheme (high, medium, and low)
- You should review this and change as appropriate
- May require a risk assessment
- What’s *high* on one type of system may be *medium* on another. System classification is therefore necessary.

System Classification by Risk

- Case Study Example
  - **Internet-facing, high risk**
  - **Intranet high risk**
  - **Intranet normal risk**
  - **Low risk**
- Problem: Nimda and Code Red/Blue made every IIS server a potential high-risk machine, so can you really exempt any machine from the process??

Mitigation Strategies (1)

- Remove all vulnerabilities in one pass
  - Not always feasible without considerable service disruption
  - Meets resistance which could erode overall success of program
  - Could work with STRONG management support
- Remove all vulnerabilities in multiple passes
  - Prioritize by vulnerability, by part of network, or business unit
  - Vulnerability prioritization: Fix systems with *highs,* then just *mediums,* then the rest in multiple passes
  - Business unit prioritization: Focus on a business unit, fix, then move on to the next one

Mitigation Strategies (2)

- Multiple Passes have problems too
  - **vulnerabilities linger longer**
  - **project may lose steam**
  - **business unit approach leaves severe vulnerabilities in other areas too long**
- There are always exceptions
  - **System is about to be upgraded, retired, moved, etc, etc**
  - **Mitigation breaks a critical application**
  - **There’s nobody to do the mitigation**
  - **Systems that cannot be mitigated must either be isolated or disconnected from the network**

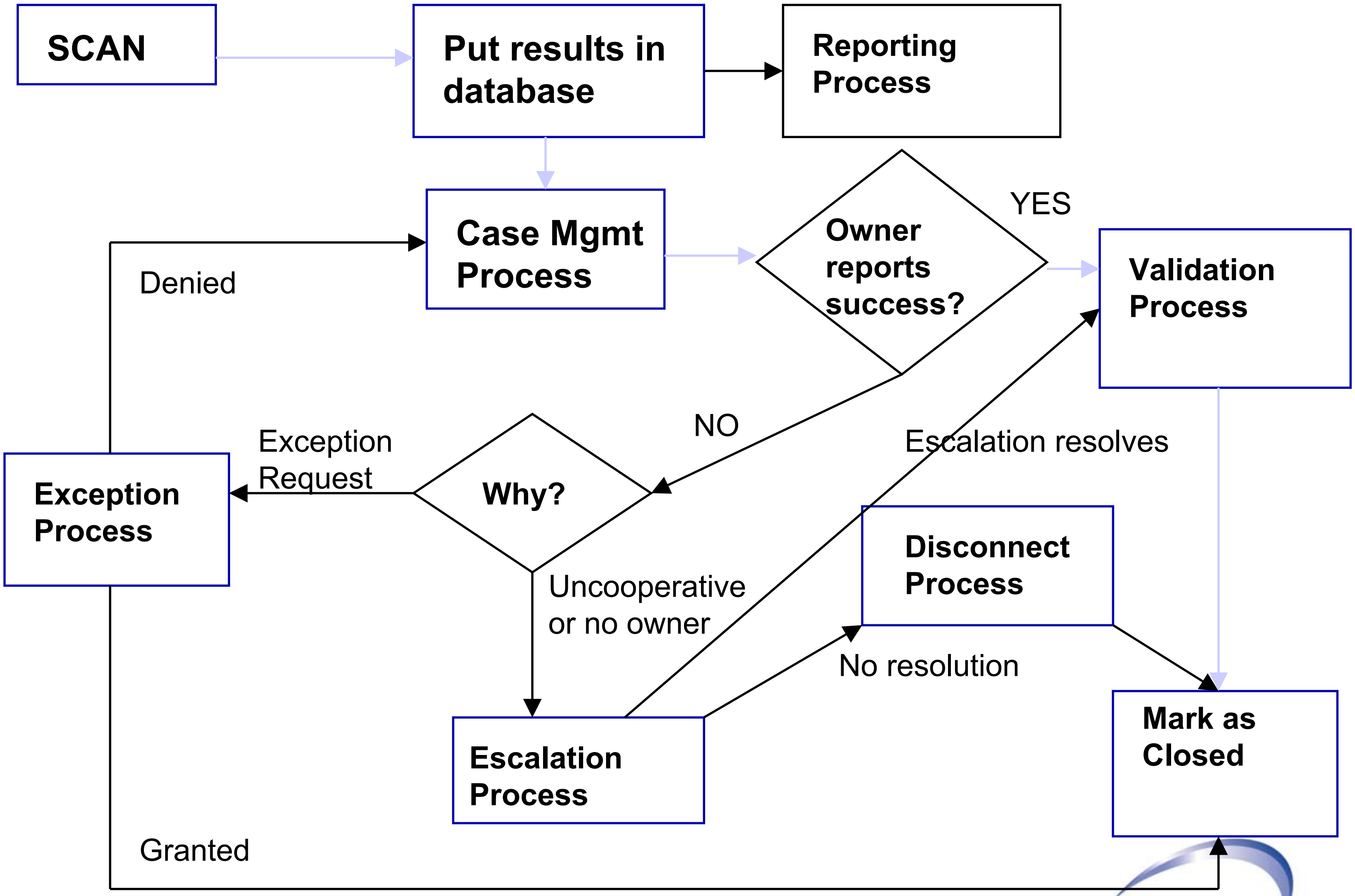
Speed, Disruption, and Effectiveness

- Your strategy must balance these 3 variables
- The speed at which you mitigate is directly proportional to the effectiveness of your mitigation strategy
  - **Greater speed reduces window of vulnerability**
  - **The slower you go, the more new vulnerabilities are discovered. You may get permanently behind**
- But speed is inversely proportional to disruption
  - **Disruption can lead to political intervention that could damage you and your project**
  - **Disruption can cause real business losses and add to the overall cost of the mitigation effort**

System Ownership Issues

- Case Study - Ownership of servers
  - **IT**
  - **Functional business Units (7)**
  - **ISPs**
  - **Marketing groups**
  - **Partnering groups, skunk works, etc**
  - **Orphans**
- Locating the person responsible for mitigation may be difficult.
- Getting that person to perform the mitigation may be even harder.
- Making sure the work was really done is essential.

Overview of the Mitigation Process



Database Maintenance Issues

- Database is absolutely essential in a large, dispersed operation
- Database support must be budgeted into the project
- Database was the primary “window” into the project for most participants
- An easy-to-use web-based application is needed. Consider making it HA.

Case Management Process

- Core process of the entire program
- Case managers placed in all major geographies
  - **Time zone issues**
  - **Language issues**
  - **Culture issues**
- Case managers who were IT knowledgeable worked best
- Case manager does following:
  - **Find owner/administrator**
  - **Contact owner and suggested fixes**
  - **Track progress and time limits**
  - **Trigger escalations and exceptions**
- Case Manager reports to Geographic Lead

Exception Process

- Managed by Business Unit Liaison
- Exception triggered by system owner request
- Process flows as follows:
  - **Exception request reviewed by Project Technical Specialists who write risk analysis**
  - **Exception request + risk analysis sent to Business Unit CIO.**
  - **If CIO approves, request + risk analysis goes to Corporate Risk Management (CRM) for analysis**
  - **CRM recommendation passed on to Corporate CIO for final approval**
  - **Rejection at any step results in return to Case Management Process**

Escalation Process

- Triggered when:
  - **time limits to mitigate are exceeded**
  - **owner cannot be found within time limits**
  - **owner does not cooperate**
- Escalation managed by Business Unit Liaison
- Uses BU chain of command to find owner or apply pressure on owner to cooperate

Disconnect Process

- Disconnect could mean:
  - disabling access to server from the Internet
  - disabling all server-initiated sessions
  - isolating the server from inside network
  - physically disabling server
- Disabling access from Internet effective for web servers
- Isolation best choice for partnering, skunk works, marketing groups
- Physical disabling necessary for non-cooperation in Code Red / Nimda cases

Disconnect Process (2)

- Disconnect triggered from failed escalation.
- Program Manager must approve.
- Notice goes to:
  - **system owner, if known**
  - **Business Unit CIO and Liaison**
  - **Network services people who will do disconnect**
  - **Help desk**
  - **Others as needed**
- 24 hours after notice, Network Services performs disconnect and reports this to Program Manager
- System marked in database as “closed”

Validation Process

- Performed by Security Specialists (see roles)
- System owner reports that required actions to mitigate have been performed
- Security Specialist rescans the system and evaluates results
- If all vulnerabilities are gone, the system is marked as “closed”
- If vulnerabilities remain, the Specialist (not the Case Manager) works with owner to fix remaining problems

ROLES

- Case Manager (already described)
- Geographic Lead
- Security Specialist
- Database support
- Business Unit Liaison
- Program Manager
- Sponsor(s)
- All roles are part-time except perhaps during start-up.

Geographic Lead

- Supervises Case Managers in a region. Reports to Program Manager
- Makes final decision as to when to escalate.
- Works with Case Managers to resolve problems that arise:
  - **Can't find owner**
  - **Owner disputes scan results**
  - **Owner demands to speak to a manager**
  - **Owner wants exception**
  - **and many more**
- Feeds back info to project to refine vulnerability rankings and improve database applications

Security Specialist

- Security expert who also has strong platform knowledge (Windows, HPUX, SunOS, etc…)
- Assists owners in mitigating vulnerabilities
- Performs the validation process
- Advises Program concerning technical decisions
- Participates in exception process

Business Unit Liaison

- Is “well-connected” in the Business Unit.
- Has enough IT knowledge to understand issues that may arise.
- Manages the escalation and exception processes.
- Represents the needs of the Business Unit to the Program

Program Manager

- Overall Responsibility for the Vulnerability Mitigation Program
- Coordinates activities among geographic regions
- Makes decisions that affect the entire Program
- Manages budgets and resources
- Reports to Sponsors

Sponsors

- Sponsors act as advocates for the Program to upper management
- Consult with Program Manager on major decisions
- Allocate resources
- Provide overall direction
- Defend the Program from political attacks or attempts to neuter it.
- Best to have a sponsor committee with representation from major stakeholder groups

Key Lessons Learned

- Good sponsors can make or break such a program.
- The database is critical and must be adequately supported
- Get the best people you can for Case Managers
- Don't confuse a Vulnerability Mitigation Program with a Corporate Information Security Program - the first is only a part of the second
- Always use processes (escalation, exception) to resolve disputes and get non-cooperaters in line. Don't get dictatorial!
- Vulnerability mitigation must be an ongoing business process, NOT a project. Once you start it, you can never stop!