Tomorrow someone will drop 0 day on your product. Will you be ready?
Microsoft Security Response Center

Security Vulnerability Response
Company-wide process
Single point of coordination and communication

Security Incident Response
Company-wide process to manage critical security threats
Quickly mobilize Microsoft resources worldwide

Ecosystem Strategy
Threat Landscape
Community Outreach and Information Sharing

Trustworthy Computing
Security Development Lifecycle (SDL)

Goals

Protect Customers by
- Reducing the number of vulnerabilities
- Reducing the severity of vulnerabilities

Key Principles

- Prescriptive yet practical approach
- Proactive – not just “looking for bugs”
- Eliminate security problems early
- Secure by design


Trustworthy Computing
Preparation: Policy and Organization
Executive Support

Steve understands the importance of security. Does your CEO?
Policy and Procedures

SDL
Internal Procedures
Vulnerability Disclosure
Business Principles
Bug Bar
Lifecycle
Release Frequency
Legal
Org Structure: The Response Center

- Response Managers
- Response Engineers
- Communications & Documentation
- Partner Liaison
- Incident Responders
Releasing Security Updates

Continuing through Release Tuesday...

Receive Vulnerability Report

Triage

Analyze the Root Cause

Variant Testing

Verification

Reproduce the Issue

Planning

Implementation

Release
Triage

Separate security from non-security
Synthesize technical details
Identify other responders and notify them

Advanced skills
Analyze malware
Reproduce the Issue

Start with the Given

- Repro Instructions
- POC on reported platform

Try to make it work

- Modify Repro Instructions
- Modify POC
- Identify the minimum repro
Analyze the Root Cause

Identify the risk

Amount of user interaction is required?
Could a user be socially engineered into the required steps?
Default vs. non-default?
How widely used is the vulnerable configuration?

When in doubt, fix the problem
Assessments of “not a bug” are often incomplete.
Planning

Set a schedule, allocate resources

If you fix bugs regularly, allocate them permanently

Identify the scope

Broader and slower vs. faster and narrower
Generally, broader is more comprehensive
Variant Testing

Find related bugs (variants)

Same code path, file format/protocol, etc
Copy-and-pasted code
Similar functions
Use code review, fuzzing, & provided repro
Implementation

Targeted
Other fixes cause bugs or regressions

Comprehensive
Fix the variant security bugs you find along the way

Good coding practices
Code reviews, change control, etc
Keep flexible
Verification

Functionality
The update fixes all attack vectors
  Usually new test modules

Regressions
No new bugs
  Don’t revert previous security fixes!
  Automated unit tests

System Integration testing
Don’t break stuff

Trustworthy Computing
Reliable update infrastructure
As few steps required as possible! Consumers don’t click.
Allow control for enterprise

Track and respond to post-release issues
Document and fix, but keep the update available
  Unless it introduces a problem that can’t be fixed e.g. data loss
Compatibility problems are slow to surface
Engineering for Response

Consider updating when building vNext

- Use a robust installer (e.g. Windows Installer)
- Test updating in beta
- Scoped, cumulative updates

Infrastructure enables response

- Trouble ticket database, information sharing systems, etc.
- Dedicated conference rooms and phone bridges
- Business continuity
Communications

Dedicated Inbound Channel (secure@Microsoft.com)

Relationships

Monitor public forums

Strategy

Principles
Consistent in:
- Terminology
- Location
- What you will and will not say

Authoritative
Helpful
- Consider both technical and non-technical audiences
- Better silent than inaccurate
- Loud about what’s important
- Use many channels to reach more users
Incident Response
Responding to Security Incidents

**SSIRP**
Software Security Incident Response Plan
Company-wide process to manage critical security threats
Mobilizes Microsoft resources worldwide

**Goals**
Quick understanding of problem
Provide customers with timely, relevant, & consistent information
Deliver tools, security updates, & other assistance
Phases of Incident Response

- Watch
- Alert & Mobilize
- Assess
- Stabilize & Recover
- Resolve
Assess the Risk

- Attacker knowledge
- Risk of attacks
- Attacker preference
- Value to attacker

Trustworthy Computing
Communicate the Risk

• Explain what’s happening
  • Internally and externally
  • Be authoritative and correct
    • *Know, don’t guess*
  • Help defenders, not attackers

• Actionable guidance: the Security Advisory
  • How do I know if I’m affected?
  • How bad is it?
  • What can I do to protect myself?
Incident Management

• There are books on this.
• Key points:
  • Centralize authority.
  • Limit the spread of information.
  • Stay calm.
  • Follow precedent where you can.
  • Follow your normal processes where they work. Throw them out where they don’t.
What Drives OOB

- Risk to customers.
  - Attack volume
  - Attack damage
  - Ability of customers to protect themselves
    - Mitigations and workarounds
    - Anti-malware and IDS signatures
- High update quality is required for release
Analysis and Other Considerations
Analysis

Identify trends
Determine deeper causes
Feed back to new product development

Trustworthy Computing
Other Considerations

Cloud
   All the eggs in one basket

Hardware/embedded
   Serviceability

Partnerships
   Legal agreements
   Good communication
   Full understanding of the problem