Automating Vulnerability Management in a Heterogeneous Enterprise

Jeff Boerio
Sr. Information Security Specialist
Information Security Management
June, 2008
Legal Notices

This presentation is for informational purposes only. INTEL MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS SUMMARY.


*Other names and brands may be claimed as the property of others.

Lego images are copyright of Lego Group.

Copyright © 2007, Intel Corporation. All rights reserved.

Last Updated: Aug 28, 2006
Introduction

• Background on vulnerability run rates
• Drivers for automation
• Discuss the vulnerability alert process
• Framework for automating the alert process
• Q & A
Vulnerability Run Rate *

Drivers for Automation

- Volume

- Accuracy, Consistency

- Speed

- Cost
Significant majority of vulnerability reports do not apply to us.

We spend almost as much time reviewing those reports as we do for vulnerabilities that do apply.
Alert Process

Receive Alert

Perform Initial Risk Assessment

Does it Apply?

Crisis?

Send to Incident Response

Queue until Full Review

Full Review Produces Formal Assessment

Crisis?

Send to Engineering Team to Patch

Record Relevant Info

Yes

No

Yes

No

Yes

No

Yes

No
Pseudocode 101

• Object Alert
  – Vendor
  – Product
  – Severity
  – Initial Rating
• Function isUsed()
• Function lookupSeverity()
• Procedure dbRecord()
• Procedure processAlert()
new Alert;
if isUsed(Alert->Product) {
    if (Alert->Severity >= HIGH_THRESHOLD) {
        dbRecord(Alert->InitialRating, HIGH);
        processAlert(Alert, HIGH);
    } else if (Alert->Severity >= MODERATE_THRESHOLD) {
        dbRecord(Alert->InitialRating, MODERATE);
        processAlert(Alert, MODERATE);
    } else {
        dbRecord(Alert->InitialRating, LOW);
        processAlert(Alert, LOW);
    }
} else {
    dbRecord(Alert->InitialRating, NA);
    processAlert(Alert, NA);
}
new Alert;
if isUsed(Alert->Product) {
    if (Alert->Severity >= lookupSeverity(Alert->Vendor, HIGH)) {
        dbRecord(Alert->InitialRating, HIGH);
        processAlert(Alert, HIGH);
    } else if (Alert->Severity >= lookupSeverity(Alert->Vendor, MODERATE)) {
        dbRecord(Alert->InitialRating, MODERATE);
        processAlert(Alert, MODERATE);
    } else {
        dbRecord(Alert->InitialRating, LOW);
        processAlert(Alert, LOW);
    }
} else {
    dbRecord(Alert->InitialRating, NA);
    dbRecord(Alert->InitialRating, NA);
    processAlert(Alert, NA);
}

Extra Credit: If your alert service/process offers updates, this process can be extended to cover that capability
Report Card

The automation of vulnerability assessment:

• Puts alert info in hands of engineers quicker
• Reduces number of assessment “re-rates”
• Results in less time spent reviewing data
• Implementation cost was negligible
Questions

Jeff Boerio
jeff.boerio@intel.com

Intel Corporation
23215 NE Evergreen Parkway
M/S EG1-105
Hillsboro, OR 97124