Coping with The Disappearance of Network Boundaries

2.18.05
Agenda

- A brief introduction (who am I and what do I do)
- How security barriers and roles are eroding
  - Physical
  - Logical
  - Organizational
- Terrifying examples from Real Life™
- Finding a productive approach to identifying and addressing security issues
- Promising trends and developments
Berbee in 30 Seconds or Less

Berbee People


Minneapolis

Wausau

Appleton

Madison

Milwaukee

Chicago

Grand Rapids

Detroit

Indianapolis

Cleveland

Cincinnati

Data Centers
Security Framework

Policies, Procedures & Awareness
- Policy Assessments
- Operational Framework Consulting
- Training & Consulting

Security Management
- Centralized Tool Integration
- Centralized Monitoring

Application
- Vulnerability Assessments
- Code Reviews
- Application Hardening

Data
- Authentication Management
- Identity Management
- Data Privacy

Server
- Vulnerability Assessments
- Intrusion Prevention
- Patch Management
- Anti-Virus & Anti-SPAM
- Mobile Client Security
- Server Hardening
- Authentication & Authorization

Internal Network
- Vulnerability Assessments
- Intrusion Detection
- Wireless Design Consulting
- Intrusion Prevention
- Authentication & Authorization

Private

Public

Perimeter
- Vulnerability Assessments
- Firewalls & Proxies
- Intrusion Detection
- VPN Remote Access
How I Pass the Time…

- Assessment
- Configuration Review
- Incident response, forensics, breach recovery
- Consulting
  - Product evaluation, needs analysis
  - Creation of custom solutions when off-the-shelf won’t do
  - Training, public speaking
- How should we prioritize
  - Spending
  - Effort
  - Resource allocation
  - ...on security?
- Help customers make better informed decisions about managing risk
The Network Perimeter

- Start with a simple case
  - Border router
  - Firewall
  - Switch
- Four networks
  - To ISP
  - Outside
  - DMZ
  - Inside
- Why do we have a DMZ?
Attacks on Switches

- Compromised server in DMZ
- Denial-of-Service (boring)
- MAC flooding: the “fail-open” problem
- Trunk protocol spoofing
- Is DMZ a good Defense-in-depth?
- What device is the network boundary in this case?
Where Does Our Stuff End?

- We don’t trust anything outside our firewall
- The firewall is where we enforce policy about inappropriate network traffic
- We think of a single device as being our perimeter
VPNs and Extruded Networks

- Hosts outside the firewall trusted, at least partly
  - Personal Firewalls
  - Split-Tunnel
- Our perimeter is distributed over many devices
- We may not own these hosts, or control them at all
Now Let’s Add WLANs In...

- Physical access is no longer an obstacle
Now Let’s Add WLANs In...

- Physical access is no longer an obstacle
- Points of ingress behind our firewall
  - War-drivers
  - Stowaways
Now Let’s Add WLANs In...

- Physical access is no longer an obstacle
- Points of ingress behind our firewall
  - War-drivers
  - Stowaways
- Can be a risk even if we don’t want wireless in house
  - Rogue access points
  - Vendors, home users
- Where is our network boundary, and who is in charge of it?
Behold! The simplest possible case!

- A user types in a simple URL into a browser (http://10.1.1.1/index.jsp)
- The request gets onto the network
- The server gets the request, and sends a response...
Side 1: The Server

- Custom Application
- Content Serving Software
- Operating System
- Network Protocol
- Network Card

JSP Files
Tomcat
AIX
TCP/IP
Ethernet
Side 1: The Server – Traditional Roles

- The Developers
- The Server Folks
- Networking Goons

- .JSP Files
- Tomcat
- AIX
- TCP/IP
- Ethernet
Side 2: The Client

- HTML
- IE
- Windows 2000
- TCP/IP
- Ethernet

- Form data, etc.
- The client software (web browser)
- Operating System
- Network Protocol
- Network Card
Who the heck knows?
- User?
- Hacker?
- Automated worm?

HTML
IE
Windows 2000
TCP/IP
Ethernet
Client-Server Interaction: We Think in Terms of Isolation

HTML <--> .JSP Files
IE <--> Tomcat
Windows 2000 <--> AIX
TCP/IP <--> TCP/IP
Ethernet <--> Ethernet
But really, it’s not like that…

- The things we’re trying to accomplish with our network (providing services) involve many layers acting in harmony
- Transactions flow up and down a stack, and we don’t control all the layers
- It’s important to consider the security of each element, and to be clear about which ones have to be secure in order to deliver our service acceptably
The Ideal Situation…

- The User
- Web server patched against HTTP sploits
- Client Software
- App exposes only predefined public functionality
- Internet
- Firewall blocks unwanted traffic
- Content Server
- DB responds only to proper requests
- Business App
- Data
- Assets
Using an App to Attack Users

Please enter your login information.

Login: 
Password: 
Log In

© 2009 Berbee Information Networks Corp. All rights reserved. No part of the software or material contained in it may be transmitted, used, reproduced, or disclosed outside of the receiving party without the express written consent of Berbee Information Networks Corp. This site contains confidential information. Unauthorized access is prohibited.

Berbee...putting the e in business™
We’re having a problem with the employee stock option web site. You might need to reset your password.

Click [this page to reset it.]

Here
What do we get?

Who is in charge of protecting the users of a publicly-available application from abuse?
“Infrastructure” vs. “Code”

// Check if user is already logged in, check for both the
// session attribute and the application cookie. If the
// session attribute is null, then set this to the
// application cookie, if both cookies are not null but
// different the user will be logged off.
String username = (String)
    req.getSession().getAttribute("portal.username");
String cookieUserName = AppRoutines.verifyCookie(req, res, 60);
if (username == null) {
    username = cookieUserName;
    if (username != null) {
        req.getSession().setAttribute(ePortalConstants.EMP_NUM_ATTR_NAME, username);
    }
}

Do we blame developers, or networking people for
the gap in communication here?
Multi-layer problems...

```c
char infile[80], username[40], mail_file[40],
   current_user[40], tmpstr[40];

/* snip of some intervening code that doesn't
   pertain to this example
   ...
   * /

strcpy(current_user, getenv("LOGNAME"));
```

- The code is bad, but the problem isn’t just the code
- It’s the compiled code, running setuid, owned by 0
- Even that might not be so bad, but we’re on AIX
- Who is in charge of predicting where/how code will run?
New Demands on Networks

- Convergent technologies:
  - IP Telephony
  - Network Attached Storage
- Capacity: these services take place in addition to regular network loads
- Performance: live media operations and fast I/O require predictable low latency and high throughput
- Features: multicast, QoS, line power, VLAN trunking, port security, identity management
- Compliance: corporate governance, internal audit, and new laws: HIPAA, GLBA, SOX...
The Big Picture

- Security is usually pretty bad
- For some reason, this is considered normal
  - We are trained to expect software flaws
  - EULA: both pieces, plus you’re liable

- There’s a trade-off between:
  - Security
  - Cost
  - Functionality (features, performance, ease of use)
Product ≠ Security

- We are tempted to think of our firewall as the place where we enforce our security rules... but...
- Most attacks on web servers occur over ports that **MUST** remain open
- So we install an IDS (say, on the DMZ) to inspect traffic for attack signatures... but...
- The IDS can’t see attacks that happen over SSL
- So we buy an SSL accelerator to decrypt traffic before the IDS sees it, or we install HIPS... but...
- What if our application misbehaves?
Challenges to Security Efforts

- Misguided: one-size-fits-all solutions (product = security); delegation of security duties (use cases)
- Threat-specific: firewalls, then operating systems, now application-layer (IDS)
- Minimalist: hard to demonstrate ROI on the purchase of security products
- Reactionary: applied as an afterthought, to counter threats that arise
- Distributed: many organizations outsource some operations (especially Internet), and accountability is sometimes hard to assign
So, the Situation Seems Grave

- Networks are getting more complex
- Traditional defenses are getting less effective
- Most organizations aren’t prepared for the new challenges
- Demand for good security is intensifying

- Oh, my! What are we going to do?
Security Strategies: How Can We Succeed?

By approaching security challenges with the following ideas in mind:

- **Holism**: Ensuring that our security strategy is thorough
- **Threat Modeling**: Thinking clearly and in an orderly, rigorous manner about risks; academic approach
- **Inverted Security**: Selecting technical measures prudently and effectively
- **Least Privilege**: Minimizing the consequences of any given failure
- **Defense in Depth**: Making the most out the tools at hand, providing fail-safe measures

These principles help us evaluate plans, form solutions, and simplify the task of maintaining effective security.
Hard Lessons from Reactionary Efforts

- Application Design and Voice Signatures
  - Cool technology fails to meet security needs
  - Application must be redesigned
  - **In retrospect:** should have documented security requirements better

- Network Design and Rogue Employee
  - Employee creates and deploys stealthy programs to attack corporate assets
  - Lost data, need to rebuild domain, need to subdivide network
  - **In retrospect:** should have considered a means of providing an audit trail for aspects of network utilization

- Incident Response Planning vs. Real-Life Incident
  - Large, geographically distributed company has a server outage
  - Can’t determine cause, so can’t decide how to react
  - **In retrospect:** An incident response policy would have saved time, and allowed administrators to respond in a more orderly manner
Architectural Concerns

Inverted security and design: Where do we place our AAA measures, and what assumptions does this imply?
Inverted Security and ROI

- The data are often not application-specific
- Appropriate rules apply for each entity and circumstance
- By placing security on the data, we are able to expose the same data through different business apps, for different purposes
Other examples of Pulling Security Inwards

- Interest in HIPS instead of just perimeter lockdown
- Focus on patch management, configuration management
- NAC/802.1x and compartmentalized networks
- Virtual machines that safeguard against rogue activities (sandboxes)
- Development techniques that insulate against mistakes (parameterized stored procedures)
- ActiveX as a transactional security measure (lightweight HIDS/anti-virus)
- Web services as a means of exposing minimal units of functionality
Why Inverted Security Might Just Work…

- By strictly defining interfaces and pre-formatting input and output, we eliminate many causes of trouble, because we know in advance what’s legitimate.
- We free people up to concentrate on their areas of specialization by allowing layers to be more discrete.
- We focus on exposing only atomic units of functionality, which are simpler, and therefore easier to produce safely.
Predictions: Hot Topics for 2005

- Network redesigns with emphasis on segregation
  - Out-of-band management
  - Facilitation of access control
- Outsourcing of utility operations
  - Business proposition will become clearer
  - Services will become commoditized
- Software security
  - Attacks on custom applications will continue to grow more sophisticated
  - Need to train developers to cope with hostile environment
- Security of endpoints/mobile assets: clearest ROI
  - CSA for VPN, wireless users
  - EFS w/PKI
Questions?

- Design & Architecture
- Development & Tools
- Deployment/Rollout
- Testing & Assessment
- Configuration Review
- Management & Monitoring
- Policy Analysis
- Ongoing Consulting