Security in Large and Complex Online Services

Francisco S. Artés CISSP, CNE
Information Security Manager
Electronic Arts Inc.
Francisco@ea.com

EA? (But they make games?!)

- PoGo.com, largest web-based gaming network in the world.
- EA.COM store
- Ultima Online, The Sims Online, Battle Field
- 4 global data centers containing thousands of servers and over 15GB of bandwidth.
- Major peering point for all top tier providers
- Over 20,000 nodes on the WAN.
- Nearly $4,000,000,000.00 (hint: billion) in annual revenue.

What I do at EA:

- Head of Information Security.
- Review and produce security policies.
- Review new network products, networks, and corporate systems.
- Perform audits, internal hacks, SoX SIP and other compliance functions, etc.
- Investigate theft of Intellectual Property and cyber trespassing.

Build a strong foundation:

- Decide on an operating system, maybe two if necessary for online deployment.
  - Harden the Operating System.
  - Have 3rd party test the operating system
    - Local access must be tested
    - Remote access possibilities must be tested
  - Document Daemons and Application Versions
  - Create image file and save time deploying with this image. *

Keep Databases away from the public

- Nest all Database servers in DMZs and private networks. Don't rely on TCP wrappers or "localhost" mapping of the listen daemon. (Use firewalls or other remote ACLs as your primary security barrier if you have to keep the Database local to a public server.)

Hire external auditors to test applications in Alpha / Beta phase

Hire an outside company to test for SQL / XML insertion, cross-site scripting, etc. Have them validate and give them plenty of time. After all, once you go live hackers will have a nearly infinite amount of time to work on your front-end code.

Look back!

- Most all developers are busy building the next application, making things better, etc.
- Most all administrators are busy fixing things, keeping up with ACL changes, adding bandwidth and servers to the network.
- Have at least one person dedicated to reviewing what you have deployed, and thus assess your exposure. *Pay attention to this person!*

Looking Back:

- At least quarterly vulnerability assessments.
- Review upgrades like JRE, PHP, PERL, SSL, etc. Have a process to test and deploy quickly.
- Update your master image.
- Update your hardening documentation.
- Hire outside auditors to test your security.

Things not to do:

- Compilers on public servers
- Source Code on public servers
- X11 systems on public servers
- Any system daemon not used for production
- Dual home public servers where one segment is on a private network.

Security Facts:

- Quick website fixes and deployment generally are not secure or safe. Take time to have your ideas reviewed by someone who only thinks about ways to break into things. Take the time to do things correctly.
- Do not leave notes, old index files, personal info in errors, etc. They will be found.
- If it is private information do not put it on a public server. This includes data “hidden” by access restrictions such as htaccess files.
- Do not use the same passwords on various systems!

Know what is happening on your servers:

- IDS and IPS are expensive, but there are systems that need to be watched.
  - Vulnerability scanners that can point-out policy violations and deltas. E.g. non DB server running TCP-783 (MySQLd)
  - Log checking applications that email administrators. (PortSentry, Logcheck*)
  - Snort and other free systems will allow you to keep an eye on traffic and find non-normal use of your sites.

Engineer networks that plan for attacks and problems:

- Use more then one type of MTA and DNS
- Use secure DNS and setup secondary at remote co-location sites on different subnets.
- Use some sort of DoS protection for vital servers.
- Have replacement routers on-hand.
- Have spare hard drives, servers, etc. on-hand.

Check your routing!

- Network Engineers are very busy people, check that the ACL was put into running memory, better yet... Make sure it was saved too!
- Check ACLs with some tool, or even by hand, often. Make sure you blocked what you needed, and left open **only** what is needed by the customer!

Things never to do:

- Test a server on the Internet before network ACLs are in place and tested/saved.
- Test an application on the Internet before it is hardened.
- Connect any server to a network before it is hardened.
- Use generic login IDs for groups of people.
- Utilize a personal login ID for a process login. E.g. BSMITH used for backups.

Things to always do:

- Document everything!
- Change passwords every 90 days. (All passwords)
- If someone leaves the team, change all the passwords and remove that person's account from all systems.
- Use two-factor authentication for all VPN and administrator level access.
- Test everything, use experts, repeat.

READ!

- Read BugTraq and other similar security sites.
- Subscribe to some form of vulnerability announcement system.
- Produce a comprehensive, yet easy to work with, internal VAAR for developers and IT.

Lessons Learned:

- DO NOT put Alpha or Beta sites on the Internet. ACLs aren't always applied, and people will find URLs that you think are hidden.
- Mediation of process patching must take place, a delay will almost always result in damage to the site.
- Update your image file. It doesn't help to patch 1,000 servers if you are going to deploy another 100 with the same bug a week later.
- Anonymous FTP = bad idea.
- Scan often!

Thank you!

My Contact information:

Francisco@ea.com or
falcor@netassassin.com

www.netassassin.com or www.ea.com

Q & A:

- I will answer any questions that won't get me fired. (No, I really do not know when *Madden 2007* will ship, or what new features will be in it... Even if I did I couldn't tell you.)