Global Information Assurance Certification Paper

Security Assessment for LLC Corporation

Assessment completed by Daniel Robb

This paper is the result of a security-focused analysis of a company, LLC. LLC is a small trading company of about 150 employees worldwide that deals in commodities and currencies, futures, options and stocks. The network consists of approximately 200 nodes in the head office and an additional 75 in satellite offices worldwide. There are several third-party applications and data feeds used that interface with the local network. As such, there are several considerations that must be accounted for in order to understand fully the decisions that are made with respect to security. In the debate between security and convenience, (the price of either is often a lack of the other) convenience often wins out by necessity. The need for a timely exchange of information often means that inherently insecure services such as NFS and NIS must be used. The analysis focused on the main server in particular (A Sun E250 running Solaris 7 hosting NFS, NIS, DNS, Sendmail and several in-house services and applications). Many of the problems that were found on the server were also found on many of the hosts throughout the network. In general, flaws with the OS on one machine could be found on most (if not all) machines. The analysis will focus on several different points. First we will look at vulnerabilities found in the operating system, both in the installation and configuration. Then we will look at vulnerabilities posed by third-party system and network flaws, followed by an in-depth examination of the administrative practices of the LLC systems group. This will include a look into backup and disaster recovery procedures.

In terms of operating system (hereafter referred to as OS) vulnerabilities, many of the possible problems could be avoided at the install and configuration phase of new system integration. New hosts are often installed with the largest and most complete installation package available. For our predominant Solaris hosts, this would be "Full OEM plus support". While this may prove more convenient for system administrator and user alike by making most packages available from the onset, thus relieving staff of the need to come back later to add needed packages and services, the cost is opening unnecessary vulnerabilities. I found many packages relating to foreign language integration installed on several hosts throughout the network that do not need to be there. Some of the other packages I found that did not need to be there included: GUI based user interfaces on servers and workstations that explicitly use command line interfaces; web server packages on hosts which provide no web serving; PCMCIA support installed on server class hosts; and development tools installed on machines which will not be used for such purposes. This is just a sampling of the many unnecessary package installations that I found on different hosts. While many of the installations may prove harmless, they only provide more opportunities for potential intruders and also serve to create more "clutter" on the host for the administrator to sort through. Unnecessarily installed packages provide camouflage for an intruder and their tools. By installing the bare minimum OS and then adding necessities after, the administrator will find it easier to fingerprint and keep track of their system. Intruders will find that vulnerabilities available to them are much fewer.

Another major flaw in LLC's installation method is that it is network based. The systems group utilizes Sun's Jumpstart feature to automate the installation of the OS on

new hosts. The benefit to the systems group is two-fold. First, multiple systems can have operating systems installed simultaneously and without administrator intervention. This is extremely convenient to those tasked with the installation of systems. Second, installations are uniform. The OS installed on systems can remain static each time. Once you get a good "footprint" for a system, you can duplicate it time after time. In the case of LLC, it was frustration with the wildly different installations performed by inadequate systems administrators that led to Jumpstart implementation. Countless systems would be installed incorrectly the same way time after time. Frustration with dysfunctional systems and wasted hours drove LLC to implement automated installations. One problem with the jumpstart program is that it utilizes lots of set-UID root programs to complete its job. Another problem with automated installations is that the first thing done to a new system is to connect it to the network when that is the absolute last thing that should be done[i]. An intruder who has compromised a network can attempt a number of ways to corrupt installations. One method (there are others) is to impersonate the install server. The first step in the jumpstart process is for the host to broadcast its Ethernet address and request both its own network information and the network information of the install server. If an intruder can spoof the address of the install server before the real one can respond, then the intruder (if proper planning went into creating a working false boot server) can install their own version of the OS, complete with Trojan horses, back doors, root kits and anything else the intruder saw fit to install. Systems are installed onto the network neatly and fully compromised. Although automated installations are very convenient, they are atrocious when it come to security and the staff at LLC should seriously reconsider their installation policies.

Another major OS vulnerability uncovered involved the patch administration of the systems. The methods by which intruders breach systems and networks are often well-documented vulnerabilities, which have been in existence for a long time. These vulnerabilities are often fixable by installing a simple patch onto the system. It is the failure to do so that keeps many intruders in business. This is why it is so vitally important that patches be maintained to the highest degree at all times. Utilizing the Sun's PatchDiag tool, I discovered that many of the hosts on the network were very outdated in their patches. Some of the missing patches were fairly critical security patches. The reason for the lack of patch updates on the systems seems to stem from a lack of communication and organization amongst the administration staff. Once the oversight was reported, it took some time and planning to brings the hosts up to date in an automated fashion. Although the patch process does not need to be automated, it seems to facilitate the timely update of systems on a network.

There are many different methods for automating the patch process. The administrators of LLC decided to write their own scripts to automate the process. The patchdiag.xref file, which lists the latest patches for all Sun systems by OS version, is downloaded on a daily basis. The systems are then scanned utilizing the PatchDiag tool. If any systems are found to be lacking, the needed patches are downloaded automatically from http://sunsolve.sun.com and then installed on the systems in question. Finally, an email is sent to the administration staff informing them of any changes that were made to any systems on the network. If a company decides to write their own process, care must be taken that the automated process is secure. Since the process must invariably run with privilege at some point (/usr/sbin/patchadd must be run as root) the security of the

automation scripts themselves must be thoroughly examined for any possible flaws before being implemented. Failure to do so could result in an intruder replacing your automation process with their malicious one. Also, checksums should be verified to ensure that the files downloaded from sunsolve.sun.com have not been tampered with.

As important as the installation process is in security, the configuration of that system is equally important, if not more so. Special care must be given to the boot process. Specifically, the /erc/rc*.d scripts should be modified to start only those processes which are actually needed. It is recommended that only the /etc/rc2.d directory be populated with the required startup and shutdown scripts[ii]. Also all NFS and RPC related links should be renamed as well. Obviously, actions on these last two points will depend on local policy regarding NFS and RPC. This is one of the biggest issues in the security versus convenience debate. The machines on our network had no services disabled. Startup and shutdown scripts populated all /etc/rc*.d directories. All default startup scripts plus some custom scripts added by staff were all intact. Any scripts not in use should be renamed, like S21perf, S47asppp, S70uucp and S30sysid.net to name just a few. Renaming the scripts is as easy as changing the S to an s.

Another major flaw in the configuration process was failure to protect the PROM level of the host by assigning it a password. Various levels of protection can be implemented, from requiring a password to input any command, to only requiring a password once to enter the PROM level and change run levels, to having no password assigned at all. A major hole exists when there is no password assigned at all. An intruder with root privileges can assigned an unknown password using "eeprom security-passwd=" command and reboot. The machine will be unusable because administrators

will be unable to boot without that password. The only way to recover from this is to install a new eeprom[iii]. Recovering from an unknown PROM level password can take weeks and requires the vendor's assistance.

Several other host-based vulnerabilities were found when integrity checks were performed on our server. These vulnerabilities were often unique to the host, but could often be found on other hosts. The following paragraphs illuminate specific examples of problems found with the main server, either in its configuration or with how it was administered. We used both Tiger (version 2.2.4) and Cops (version 1.04) to analyze the server.

Several entries are repeated multiple times in /etc/passwd. Both usernames and UIDs can be found to repeat themselves.[iv] Two usernames sharing the same UID can cause conflicts, especially on an NFS served network. There should only be one instance of any username or UID in the /etc/passwd file. This was a pilot-error type hole opened by the system administrators when adding users. Assigning a new UID to one of the usernames, altering the ownership and making copies of any files belonging to that user solved the problem. The administrators were made aware of the holes and told to take better care when assigning UIDs and usernames to new users.

Several user accounts that had been disabled were found to still have valid shells assigned to them.[v] Although the account is disabled by altering the account's entry in the /etc/passwd field, an extra layer of protection against its unauthorized use can be implemented by assigning it a non-shell, such as /bin/false.

There were several instances of directory and file permissions that opened unnecessary holes in the system. Directories that contained scripts executed with root

permissions in the crontab were writable by ordinary user groups.[vi] An examination of the permissioning of files and directories ought to be run regularly to check for anomalous or dangerous file permissions.

Because of the nature of LLC's business, the network plays a crucial role. The timeliness of data transfer makes the difference between making lots of money and losing lots of money. The LLC network interfaces with many different third parties. Real-time financial data is constantly fed into the network, where it interfaces with both third-party and proprietary applications. The major threat from these third-party interactions is the level of trust and openness that sometimes has to be maintained between vendor and client. Third-party routers and servers are placed on the network physically and there is no real accounting for how that particular company maintains their security. To help combat a possible intrusion through these third-party systems, a DMZ should be appended to LLC's firewall structure, which contains all traffic connected to the Internet and to any outside third-party services so that access is more strictly regulated and logged.

Other third-party applications seem to be maintained in an appropriately secure fashion. LLC is running the latest version of Sendmail, V.8.9.3.[vii] Relaying is disabled to prevent spamming from our site. The one improvement that could be made would be to change the Sendmail banner to something less informative. As you can see from the output listed in the footnote above, the version of Sendmail running is listed to anyone who can telnet to port 25. It would be a good idea to change the banner to something a little less revealing. The sendmail.cf file on the mailhost needs to have a

"define('confSMTP_LOGIN_MSG'," string added[viii]. This will confuse would be attackers, especially if you list an older version.

DNS and BIND are running at 8.1.2. Some DNS and BIND vulnerabilities are accounted for and disabled though some major ones are not. One major vulnerability, which is not defended against, is unauthorized zone transfers. There is no 'allow-transfer' substatement listed in the /etc/named.conf[ix]. This means that any intruder can force zone-transfer to a spoofed slave server. This will provide the intruder with a wealth of information about the network. LLC's implementation of BIND is not utilizing the split-horizion feature which differentiates between the internal and external networks and delivers scaled information depending on where the request is coming from. Those who utilize the split-horizion feature are sometimes not as concerned about unauthorized zone-transfers because the information delivered through the zone transfer is considerably scaled down. A third feature of Unix which ought to be implemented is to run LLC's DNS and BIND in a chrooted environment. Although it is listed last, it would be perhaps the most effective course of action in helping to secure the network against any possible BIND attacks.

Other network based vulnerabilities implemented on the network exist in the utilization of NFS and NIS, and RPC. Although there are many dangerous flaws in these programs, their usefulness to LLC may outweigh any decision to change the way things are done. One of these flaws in particular that can be improved without adversely affecting the performance of the network is the use of RPC. LLC uses all of the Remote Procedure Calls in their original, unadulterated form. Data transmitted between nodes on the network using these methods can be gleaned using a packet sniffer and then used to

the advantage of the attacker. This is a very insecure medium for data transfer. There is a simple solution for this, which should be implemented right away. That is to use SSH, which can be dropped into the place of the RPC commands. The SSH variety of RPC commands perform the exact same functions but encrypt all data moving across the network so that anything captured enroute is practically useless. This way, even if the network is compromised, anything stolen from it will not go to malicious use. Another friendly feature of SSH is that its binaries can replace those of the RPC commands, making the enhancement transparent to the users[x]. SSH can also be implemented with TCP-Wrappers, which will increase logging usefulness on the network.

The administrative practices of the systems and networks staff, while adequate, (given the resources available to the group and the load placed upon them), could be easily enhanced with some minor changes. First of all, logging needs to be increased. Authentication messages should be logged to their own file instead of displayed on the system console, as is the current practice. Cron logging should also be set to log to specific locations. Currently, the syslog is the only log that is rotated so that data is maintained for a longer period of time. The authlog should also maintain data for a few weeks before being either discarded or rotated onto removable media. Ideally, it would be best to rotate logs on a weekly basis and then compress and store all rotated log files on a monthly or quarterly basis. Compressed logs can be stored on tape and then placed in a secure location off premises. Once more data needs is collected, logging programs and be installed to maintain a better awareness of what is happening on the network. Because of the volume of data that could be logged, sifting through the logs needs to be automated. Using a program like Swatch or Logcheck can help administrators to track

what is happening on the network. Reports of log activity can then be mailed to administration staff so that action can be considered.

Auditing of the file system's integrity needs to be expanded. All systems need to be fingerprinted and then scanned often for changes in the fingerprints of the system. The Tripwire program is designed for this type of task. Files that are to be checked by Tripwire are listed in the tw.config file[xi]. It is extremely important that when the initial database of signatures for scanned files is created, it is created from known secure files. This means that system files are thoroughly checked first. Ideally, Tripwire should be made part of the installation process where images of the files are taken right after their installation. If you are installing securely and not off of the network, then the administrator can be reasonably sure that the files on the machine have not been tampered with in any way and that the signatures of those files are genuine and valid. Tripwire should be run several times throughout the day.

Password management needs to be improved. The encrypted passwords need to be run against a password cracker on a regular basis. This was not being done before this audit was performed. I installed a password-cracking program (Crack V.5.0). I used the output from 'ypcat passwd' for the input for Crack. The results indicate that stricter password controls are needed. 10% of the passwords which were run through Crack were successfully guessed. Some include passwords such as d1sney and m0n3y although to the users credit, there were none that used their username as their password. Especially if LLC is going to use NIS where anyone on the network can run ypcat to get a copy of all the encrypted passwords, then certainly, passwords protection need to be stricter. Specifically, crack needs to be run often to make sure that the integrity of the

passwords is maintained. Also, password aging needs to be implemented along with new password checking so that passwords are changed every few months and replaced with secure ones.

The process of backing up data at LLC is fairly sound. A full level 0 dump of all relevant file systems is performed over the weekend when the network is relatively quiet. Level 0 dumps are placed compressed on a DLT tape attached to the main server. Level 0 dumps are kept in the server room for one week and then placed in a safe-deposit box in a nearby bank vault. The first level 0 dump of each month is kept for two years. The other weeks' tapes are recycled after six months in the vault. Only the partners of the firm and the systems staff have access to the vault. Access to the vault is verified by possession of the key to the box (held by one systems administrator and one senior level partner) and by verifying the signature of the person entering off a signature card on file. In addition to the weekly level 0 dump, level 5 incremental dumps are run nightly during the week. These back up whatever has been changes since the last dump occurred. Two weeks of incremental dumps (one 8mm tape for every week day) are kept on site in the server room. Only these two weeks of incremental dumps are kept.

The major security flaw uncovered in the backup scheme employed by LLC was that some backups were performed over the network. In addition to the major file systems on the main server being backed up, all research and development staff have local /export directories where their work is kept. These /export directories are backed up over the network. The inherent problem with performing backups over the network is that an intruder could intercept the data enroute to the tape device. They could steal this data for their own use (LLC has many proprietary in house applications that could be

valuable to others) or they could alter, corrupt or destroy the data. A more secure solution would be to place removable media drives onto all workstations which require backing up, such as the Iomega Jaz or Zip type drive. It would be an improvement to the security of the backups in that no data being backed up would pass through the network. A downside to this however would be in the increased cost of purchasing, installing and supporting these drives. Approximately 20 workstations needing zip-type drives at $140/per workstation (http://www.cdw.com) plus staff hours to install and configure plus $100 for media per workstation comes out to approximately $5000.00 (labor charges are site specific obviously). Another issue to consider is that the backups of individual workstations is now scattered amongst many different tapes instead of just one tape.

One area in which LLC is dangerously unprepared is in its disaster recovery scheme. LLC is adequately diligent when it comes to keeping spare hardware on the premises. If a major physical or environmental incident were to occur at their main office, the company would face a severe operational crisis. The nearest satellite office is about 3000 miles away. LLC needs to implement a more comprehensive disaster recovery plan. An alternate location needs to be found to house the disaster recovery site. The site should be far enough away from the main location that even if the disaster event were large enough to damage several city blocks or miles even, the backup site would still be operational. It should not be so far away that it takes an appreciable amount of time to get to the backup site and resume operation. I would say that a location five to ten miles away from the main site should suffice. The backup site should be served by a power grid and phone company central office different from those of the main site. The backup site should have power and telecommunications features similar to the main site.

Multiple power circuits should come into the building to provide redundancy to the UPSs.

In LLCs case, the backup site should contain at least one server, which will provide all the necessary file sharing and network services and about ten workstations for traders to use in the execution of their duties. Multiple phone lines will need to be installed (preferably in hunt-group style for convenience) for communication with the outside world. Some duplicate data feeds will need to be installed to feed the trading programs running on the server. One or two third-party services will also need to be connected so that trades can be executed. All staff provided cell phones by the company should have them at all times so that they too can be used in a disaster recovery situation. The cost of all this is not cheap by any means. There are ways to make it less of a drain on the company. For LLC, the disaster recovery site is going to be located in the basement a senior partner's house. This will relieve LLC of the need to pay rent and possibly collocation fees for their backup location. There will still be one time setup costs (hardware: servers, workstations, UPS, switches, cabling, routers, etc) and also monthly recurring costs (telecommunications and vendor monthly charges). Although the project will not be cheap, it is definitely justifiable given the alternative.

Physical security at LLC, with the exception of one or two areas, is good. LLC is on the 26th floor of a building in a metropolitan center. The building is accessible 24 hours a day, seven days a week. This can be a big benefit when it comes to telecommunications repairs. There are two entrances into the building. The front entrance faces the main street and has a guard posted at a security desk at all times. The other entrance is the loading dock off the alley behind the building. This has a guard

posted during business hours only. That entrance is locked down (going into the building) when no guard is present. Access to the building is free during business hours without having to account for ones presence. Between 6pm and 6am, people entering or leaving the building must sign in and out with security. Security personnel inspect any packages leaving the building. In addition, a package-removal form signed by an authorized representative of the company must accompany all packages. Surveillance cameras are located throughout the lobby and in the elevators.

I tested the security of the building several times over a few weeks. The security staff did well for the most part, but I was able to circumvent security on a few occasions. One time I was able to leave through the loading dock after hours without signing out. Since I entered the building before 6pm, there was no record of my presence there after hours. It would have been a easy matter to carry a package out, as long as it did not significantly burden me. I also once caught the night guard asleep at his station. I was able to leave the building through the front door without being challenged at all.

LLC owns most of the 26th floor. There are three entrances into the office space: one at reception, one to the break room, and a third into the hardware assembly area for the systems group. Proximity sensors and keycards protect all three doors. An additional two doors are also protected: the door to the server room and the door to the telecommunications room. All employees have access to reception and the break room. Only systems staff and the senior partners have access to the other three areas. Strict logging notes who deactivates any lock with a timestamp. Also, only systems staff, senior partners and the night traders are granted access to the office outside the hours of 6am to 6pm, Monday through Friday. This means that there are only about 10 people

who could possibly gain physical access to the servers or routers, unless one of those people had a card lost or stolen and failed to report it. There are raised floors connecting unsecured and secure areas, but the floor panels are screwed down from above so while it would be easy to get down there, it would be difficult to get back up. Both the server room and the telecommunications room are environmentally controlled for both humidity and temperature with backup system in place. All systems are on UPSs which are all individually fed by separate circuits. The UPS situation in the server room needs to be readdressed since it appears that all UPSs are running at maximum capacity. One problem is that the fire suppression systems in the server and telecommunications rooms are water based. Obviously this was a rather large planning oversight. Not only would water aggravate an electrical fire, but it would also damage every other piece of equipment in there.

My overall assessment is that LLC's security policies need work. Administrators in the past have been too busy to try and implement the daunting task of securing an entire network from scratch. This lax policy has begun to get LLC into trouble. The Systems administrators need to change their policies relating to how security is dealt with. Tools need to be implemented to examine the network in great detail and then automated so that security administration need not be a full-time job. With work, the systems and network can be fully secured. Then the processes by which the systems and network will remain secure (or at lease detected if otherwise) can happen without much user intervention. All of the problems I have outlines here today need to be resolved as soon as possible. Some are more important than others. I will start with those.

The integration of eeprom passwords must be completed. If an intruder were to gain root access to the right server, one could set the PROM password on all machines, and render all of LLC's servers and workstations useless. This exploit would put LLC out of business. The RPC based services need to be replaced with their secure SSH equivalents. The installation of machines needs to not be network based. If the administrators of LLC cannot live without that convenience, then the jumpstart process must be performed on a separate network that contains only the install server and the client. The configuration of the hosts needs to be revised, with countless unneeded services turned off by default. The patching of vulnerable systems needs to be addressed in a more timely fashion to prevent unnecessary flaws from being exploited. File system integrity and logging need to be overhauled. Without this, LLC staff will never if and what hit them. Scans (COPS, Tiger, Satan, nmap) need to be automated, logging (scan logs with Swatch and Logcheck) and general awareness increased. Passwords need to be scrutinized. Systems staff need to be more careful when altering critical files like /etc/passwd. Making mistakes like assigning shared UIDs to multiple users will reek havoc eventually. Finally, a comprehensive disaster recovery plan needs to be implemented so that if disaster ever strikes, LLC will not be out of business. All of these issues are absolutely crucial to LLC's future well-being and prosperity.

---

[i] From "Solaris Practicum" by Hal Pomeranz, p.13
[ii] From "Solaris Security Step by Step" The Sans Institute
[iii] From "Solaris Practicum" by Hal Pomeranz, p.82
[iv] taken from output of Tiger
--WARN-- [pass001w] Username `joeuser' exists multiple times in /etc/passwd.
--WARN-- [pass002w] UID 1026 exists multiple times in /etc/passwd.
--WARN-- [pass002w] UID 1035 exists multiple times in /etc/passwd.
--WARN-- [pass002w] UID 1038 exists multiple times in /etc/passwd.
--WARN-- [pass002w] UID 30001 exists multiple times in /etc/passwd.
[v] --INFO-- [acc002i] Login ID danw is disabled, and has a shell of /bin/ksh
[vi] --WARN—[cron003] cron entry for user uses

'opt/share/scripts/script_example' which contains '/opt/share/scripts'
which is group 'develop' writable.

[vii] From output of "telnet server 25"
220 server.llc.com ESMTP Sendmail 8.9.3+Sun/8.9.3; Mon, 30 Jul 2000 18:18:42 –0500 (CDT)

[viii] From "Running Unix Applications Securely" By Lee Brotzman and Hal Pomeranz, p. 128

[ix] From "Running Unix Applications Securely" By Lee Brotzman and Hal Pomeranz, p. 82
Also from "DNS and BIND" O'Reilly and Assoc. p.252

[x] From "The Secure Shell (SSH" By Steve Acheson

[xi] From "Unix Security Tools and Their Uses" By Matt Bishop, slide 118