Global Information Assurance Certification Paper

When Uptime is Critical: Solaris Recommended Cluster Installation in a Geographically-Diverse Enterprise

Stephen Gracon
GSEC Assignment Version 1.4b – Option 2
15 Jun 2003

Abstract:
Keeping servers current with the latest patches and maintenance updates can be a difficult task for systems administrators. Most servers require 24/7 uptime and provide mission-critical services (such as DNS, NFS, and NIS[1]) so there must be a fine balance between server uptime and security patching. Many enterprise-class companies have service level agreements (SLAs[2]) which limit the time and length of any acceptable downtime further restricting the how and when a server may be patched. This creates a daunting task for the systems administrator; they must balance the needs of security with the requirements of their clients and own organization.

This issue was presented to me by the Vice President of Technical Operations. Due to some unknown factor, information security for my company went from a thought in the back of everyone's mind to a 1A priority. In one months time I was tasked to prepare, test, and implement a solution to patch 82 enterprise-class Sun Sparc servers, in various states of repair, and do so with minimal downtime while still adhering to the SLAs that are established with our clients.

Solaris Patching:

While there are many different avenues to pursue when patching a server running Sun Solaris[3], most times the simplest approach is the best. For this Sun Microsystems has created the "Recommended Cluster" for each version of the operating system as well as for each hardware platform, both SPARC and x86. Contained in the rather large ZIP file is each patch recommended by Sun for installation on the operating system as well as an automated script for one command execution. The ZIP file also includes a file detailing the proper patch application order.

Per the included README file, Sun highly recommends that the recommended patches be installed with the server in Single-User (Maintenance) Mode. While a simple operation with a non-production system local to the systems

[1] Domain Name Services (DNS), Network File Services (NFS), and Network Information Services (NIS) are critical to an enterprise as they provide host name resolution, network file sharing, and user authentication respectively. An entire system infrastructure can cease to work if one or more of these were to go down at an unscheduled time.
[2] Service Level Agreements (SLA) are put in place between two companies as a measure of specific performance such as data center uptime, acceptable maintenance periods, etc. There is typically a penalty and/or a reward involved depending on compliance.
[3] Various patching methods are available at both http://sunsolve.sun.com and http://www.sun.com/bigadmin/

administrator, Single-User Mode creates the following issues for production/remote servers:

1. Network services, such as NFS, are unavailable as they are started in Run Level 2 and 3
2. /etc/nologin is created automatically denying all remote log-ins. Only local access is granted.

The recommended cluster installation process adds a few more issues to the mix as well:

1. Depending on the period since the server was last patched, the process may take a few minutes to an hour or more.
2. The cluster is best run from the local system and therefore requires space for the ZIP file (50-85MB at present) as well as the unzipped contents

These issues will be addressed later in this paper as they are a critical stop point when installing the recommended cluster across geographically scattered servers.

Planning:
Without proper planning any project can be guaranteed to fail or at least fail to meet all predefined objectives (assuming there were some). While one can never plan for every contingency, writing processes down on paper and having others review them can save the creator many hours of sleep. With that in mind I created as list of factual information from which I could identify how to proceed.

Management Request: Keep all enterprise-class servers current with the latest security patches and maintenance updates while adhering to the guidelines in our client's SLAs. This needs to be completed in 30 days or less.

SLA Requirements:

1. All maintenance must be performed between the hours of midnight and 5:30a.m. Pacific Standard Time Monday through Thursday.
2. 24/7 operation with no primary and redundant system being down at the same time.

Recommended Cluster Requirements:

1. Approximately 400MB free for both the recommended ZIP file and the contents of said file.
2. It is highly recommended that the system be in Single-User Mode, therefore it is best to make this a need.

3. The cluster must be installed as the Super-User or equivalent[4].

My (Security-focused) Requirements:

1. Enable the Network Operations Center (NOC) personnel to install the cluster without giving them direct access to the Super-User account (i.e. without them knowing the root password).
2. Use encrypted methods to communicate with each server.
3. Securely transfer the recommended cluster to each server as well as store the cluster ZIP file in a uniform location.
4. Remove the ZIP file and associated patches after installation has completed successfully.

These are the initial requirements. They are the hard fact, 30,000' overview needs of the project. With them defined the project now has direction and a source for better refinement of needs and possible issues.

SLA Requirements:

As my company operates in a 24/7 environment, we have coordinated with our clients to create a specific maintenance "window" during the period of least traffic. This way any outage that may be catastrophic in nature will cause the least amount of loss to both our clients and our company. This window is only available Monday through Thursday since client traffic is significant enough Friday through Sunday to cause a hardship if a failure were to occur. Server patching issues can occur if the NOC is unable to complete the patching during the maintenance window.

Each of our 41 remote locations has a primary and secondary server. This provides some redundancy if one of the servers requires maintenance or fails. Only one server at a time can be brought down at a time for patching which increases the length of time required to patch a site as well as diminishes the redundancy of the site.

Recommended Cluster Requirements:

Many of the servers at each remote site were state-of-the-art when they were installed, but that was several years ago when storage space was smaller and more expensive. The generic storage location, /var/tmp, may or may not be large enough to accommodate the 400+ MB requirements for the recommended cluster. Even if there is adequate space available, there is the possibility that other temporary files exist in that location creating a different lack of space issue.

[4] The person who has Super-User (root) access to a server has the metaphorical "keys to the kingdom" and can do just about anything they wish. This is why Super-User access is closely guarded by systems administrators.

By default, Single-User Mode is only accessible via a console or keyboard/video/mouse (KVM) source attached directly to the server. While they may be attached to the remote server, there is no technical staff on-site and the root password is not something that any systems administrator will give out freely to the non-technical user at the remote site during the maintenance window (or at any time actually)[5]. The patching methodology must include a way that allows our centrally located administrators to access each server in Single-User Mode.

Only our systems group has the ability to log-in to a server directly as the Super-User. Other users, such as our NOC staff and developers are granted root-like access through the use of the `sudo` (Super-User Do) command[6]. This command would give the NOC the ability to run the cluster executable as the Super-User as well as logging when they executed the command. This command also has the added benefit of not requiring a member of the systems group to initiate every cluster install during the maintenance window.

My Requirements:

Thanks to `sudo`, my first requirement can be solved with minimal work since the infrastructure already exists. The NOC is already granted permissions to install applications during the maintenance window so the only need is to add the absolute path of the cluster install script to the sudo configuration file[7].

The next requirement is more difficult than most. While all servers have Secure Shell (SSH) installed to encrypt user sessions, changing run levels from fully-operational (Run Level 3) to Single-User Mode (Run Level S) will terminate all open sessions. The system also creates the /etc/nologin file which refuses all remote connections to the server. Careful thought must be used when deciding how to enable the NOC to remotely connect to a server that is in Single-User Mode.

Since SSH is installed across the enterprise, we can use the `scp` (secure copy) sub-command to securely copy the recommended cluster from a known-good location to each server. Additionally, this process can be automated through the use of SSH key management and scripting.

More of a checklist item, the NOC must make it a habit to remove the cluster ZIP file as well as the associated directory after successful cluster installation. It

[5] Systems administrators are expensive and it would be a very large sum of money to employ an administrator at each of the 41 remote sites. That is why all servers are administered remotely from a central location.

[6] Sudo is an open source program, maintained by Todd Miller, which enables normal users to execute commands as the Super-User. The user's actions are limited and are logged for audit purposes. More information can be found at http://www.courtesan.com/sudo/.

[7] The absolute path is used (i.e. /var/tmp/8_Recommended/install_cluster) instead of the relative path (i.e. ./install_cluster) to reduce the possibility that the user will create a malicious file called install_cluster and execute it as root.

would also be a good time to verify that the system has rebooted successfully and has returned to Run Level 3 (full operation).

Preparation:
With the requirements clearly defined, we can now create dependencies as well as specify software and training requirements to ensure this project's success. First we will tackle the software and operating systems needs, then the training required for both systems administration staff and NOC personnel.

Software:
As a policy, my company only installs programs and executables in Solaris package format[8]. This enables quick access to what programs are installed as well as an easy and complete way to remove them if needed. The following packages are required for successful completion of this project:

1. Sudo: Sudo is already installed across our enterprise, but every configuration file needs to be verified that they grant the correct permissions to the NOC and other personnel. Each server that requires patching performs the same function so we can implement a common configuration file across all servers
2. Secure Shell (SSH): As with Sudo, SSH is already installed on every server to enable secure communications. We need to ensure that the SSH daemon is started and running in Run Level 2 and is not terminated when the system is shifted from Run Level 3 to Single-User Mode (Run Level S).
3. /etc/nologin: In order for our NOC personnel to log into the server, we need to remove this file every time the system is shifted to Single-User Mode. To solve this issue I created a package containing a script which does just that. Every time the server processes the start scripts in /etc/rcS.d, the /etc/nologin file will be removed enabling users to log-in to the server.
4. [6,7,8]_Recommended.zip: Sun updates their recommended packages regularly every week or two and they are publicly available at http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access. The method I choose is one of routine. Every Monday I retrieve the latest recommended packages and compare their MD5[9] signatures to verify if there has been a new release. The files are then placed in a commonly accessible location, in this case on our corporate NFS server, for disbursement to each server which requires patching.

[8] Solaris packages, much like Redhat's RPM format, are used to install multiple files and folders onto a system. A record is maintained in /var/sadm/install/contents of everything installed so that it may be entirely removed at a later date. See http://docs.sun.com/db/doc/805-6338 for more information.
[9] MD5 is an algorithm used to create a hash of a file. This hash can not be duplicated unless the file is identical. More information can be found at http://userpages.umbc.edu/~mabzug1/cs/md5/md5.html.

Quality Assurance:
Even though Sun performs in-depth testing on their patches, it is important to have a "burn-in" period for the recommended clusters before they are installed enterprise-wide. For this I use a three week rotation; week one the cluster is installed in one or more labs located at the corporate headquarters, week two it is installed in one remote site within driving distance to the corporate headquarters, and week three the cluster is rolled out to the rest of the remote sites.

This burn-in period is necessary not only for the stability of the servers that receive the patches, but also due to the difficulty of figuring out which patch may be interfering with a server if the cluster installation causes unwanted effects. The three step process gives two weeks for any problems to arise before the cluster is installed enterprise-wide.

File Transfer:
Each server to be patched must be checked manually to verify the location and available space for the upload of the recommended cluster. While this is a time-consuming task, it is a necessary step in ensuring that the server is ready to receive the file.

Through a home-grown set of scripts, we are able to automate the distribution of the recommended cluster via rsync over SSH[10]. While the SSH tunnel is not absolutely necessary, it reinforces good habits of transferring data and logging into systems via secure methods.

Training:
The better trained the NOC personnel are; the less likely the on-call systems administrator will receive a phone call in the middle of the night. With the KISS (Keep It Simple Stupid) principle in mind, I created three types of documentation:

1. Written instructions: A person will have a much easier time performing a task if there are written instructions in front of them and an associated checklist.
2. Electronic instructions: Along with the paper instructions, I also created an electronic copy located on the corporate intranet. Having the user utilize the "highlight and middle-click" functionality of Solaris, I can reduce any problems associated with the user fat-fingering a command.
3. On-the-job training (OJT): Installation of the cluster is to be performed by our graveyard shift NOC personnel; therefore it is in the best interest of all parties involved to walk them through patching a few

[10] To explain the rsync concept would be a paper unto itself, but for those interested in reading O'Reilly has a good article on it at http://hacks.oreilly.com/pub/h/38.

servers to answer any unforeseen problems before they happen across the entire enterprise.

Modification Approval:
It is always important to have management buy-off of any modification that will either cause an outage or has the potential to cause an outage. In my company's case, we utilize our existing Remedy[11] help desk system and have created a Technical Modification sub-form. This form automatically populates which managers are required to OK a proposed modification.

For the patching of servers, the notified parties are the Systems Group Manager, the NOC Manager (for scheduling of personnel), and the VP of Technical Operations, the ultimate authority over all things of a technical nature. It is in this Technical Modification form that I once again detail the installation instructions, expected downtime, and other fields of interest to management. The form also details the total time needed to perform the entire install across all sites. Remedy is intelligent enough to alert the submitter if their proposed installation will overrun the allotted time.

Measurement:
Since this is the first time I ever attempted this type of project, I have no real idea how long the installation process will take. With that in mind I created a few key points to monitor as the first and subsequent installations occur:

- How long does each installation take, on average, to complete
- What influence does CPU speed, RAM, disk type have on the installation
- What pitfalls occurred? Were there any unforeseen patches/packages that were required beforehand?

With this data I can establish a baseline to help determine if the installation process can be improved. I can also verify that the NOC personnel are in fact performing up to expectations. While trustworthy, if each server takes half an hour to patch and only four are completed in a five hour period; there is an issue at stake whether it be the personnel themselves or the process.

Installation:
It is at this point where all requirements should have been met. As a review, we have accomplished the following points in preparation of installation:

1. Defined and verified the installation of software required for the installation of the recommended cluster.
2. Established a quality assurance process to greatly reduce the possibility of the cluster causing server issues.

[11] Remedy Action Request System is a well known enterprise help desk suite. Details are available at http://www.remedy.com/.

3. Selected a secure method of transferring the recommended cluster from a central source to all servers to be patched.
4. Trained all personnel involved in cluster installation and potential pitfalls.
5. Received approval from management to install.
6. Established key concepts for measurement of the installation process.

Week One:
Since our lab servers are non-production, I walked several NOC personnel through the installation process. I was only there as a guide, though, as I wanted to verify the installation instructions were simple and able to be completed by someone completely untrained with little or no questions. I also streamlined my documentation during this install process.

Week Two:
With the successful installation of the recommended cluster across our labs, it was time to implement it in a single site production environment. Since this is production, this must occur during the maintenance window and is an ideal to train the graveyard NOC shift. Since this site is near by the corporate headquarters, they are also used to being the proverbial guinea pig and understand more about potential outages than the other sites.

Week Three:
Since installations in our labs and at our "local" remote site were successful, it was time to roll out the cluster to all remaining sites. Four days were allotted for installation on the 82 remaining servers. As with any project manager, my cell phone was on and I was at the ready awaiting that phone call that the cluster has caused catastrophic failures across the enterprise and the CEO wanted me to conference in and explain what was wrong. Fortunately, my phone never rang once during the four days.

One unforeseen issue did occur during the installation process. Since I am not the only administrator or developer for the company, there were other, unrelated packages that required installation during the same maintenance window as the recommended cluster. These other packages created some time management problems as critical fix application packages take precedence over recommended security patches. While information security is important, unless it is causing a current financial loss, it takes a back burner to that which is, namely a problem with our revenue-generating applications.

The final period of installation was reached and there were still a few sites whose patching was not complete. It was rescheduled for the next acceptable maintenance window and was completed, albeit later than I would have liked.

Conclusion:
While the patching of servers is a never-ending process, it can be a relatively painless process so long as there is in fact a process and forethought. This was a success due to:

1. The establishment of the project guidelines, requirements, and an overview of what we would like to accomplish.
2. Preparation of not only the core task, but all related tasks
3. Proper documentation as well as the understanding that projects are dynamic entities that always have room for refinement

All in all, this entire project was a success, and has been implemented for several months with no significant issues.

List of References:

- Abzug, Mordechai. “MD5 Homepage (unofficial).” Jun 2003. URL: http://userpages.umbc.edu/~mabzug1/cs/md5/md5.html
- BMC Software. “Remedy, a BMC Software company.” Jun 2003. URL: http://www.remedy.com
- CIO Magazine. “Putting IT in Writing – SERVICE LEVEL AGREEMENTS.” Nov 1998. URL: http://www.cio.com/archive/111598_sla.html
- Cromar, Scott. “Configuration and Patch Verification on Solaris Systems.” Jan 2003. URL: http://www.sans.org/rr/paper.php?id=921
- Miller, Todd. “Sudo Main Page.” May 2003. URL: http://www.courtesan.com/sudo/
- O’Reilly Network. “O’Reilly Network: Using rsync over ssh.” Jun 2003. URL: http://hacks.oreilly.com/pub/h/38
- Sun Microsystems. “docs.sun.com: Application Packaging Developer’s Guide.” Jun 2003. URL: http://docs.sun.com/db/doc/805-6338
- Sun Microsystems. “Sun Microsystems – BigAdmin Portal.” Jun 2003. URL: http://www.sun.com/bigadmin
- Sun Microsystems. “SunSolve Home.” Jun 2003. URL: http://sunsolve.sun.com/pub-cgi/show.pl?target=home
- Sun Microsystems. “SunSolve Patch Access.” June 2003. URL: http://sunsolve.sun.com/pub-cgi/show.pl?target=patches/patch-access

Last Updated: July 22nd, 2017

Upcoming Training

SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS New York City 2017	New York City, NY	Aug 14, 2017 - Aug 19, 2017	Live Event
SANS Salt Lake City 2017	Salt Lake City, UT	Aug 14, 2017 - Aug 19, 2017	Live Event
Community SANS Omaha SEC401*	Omaha, NE	Aug 14, 2017 - Aug 19, 2017	Community SANS
Community SANS Trenton SEC401	Trenton, NJ	Aug 21, 2017 - Aug 26, 2017	Community SANS
Virginia Beach 2017 - SEC401: Security Essentials Bootcamp Style	Virginia Beach, VA	Aug 21, 2017 - Aug 26, 2017	vLive
SANS Chicago 2017	Chicago, IL	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Virginia Beach 2017	Virginia Beach, VA	Aug 21, 2017 - Sep 01, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
Community SANS Pasadena SEC401 @ NASA	Pasadena, CA	Aug 23, 2017 - Aug 30, 2017	Community SANS
Mentor Session - SEC401	Minneapolis, MN	Aug 29, 2017 - Oct 10, 2017	Mentor
SANS San Francisco Fall 2017	San Francisco, CA	Sep 05, 2017 - Sep 10, 2017	Live Event
SANS Tampa - Clearwater 2017	Clearwater, FL	Sep 05, 2017 - Sep 10, 2017	Live Event
Mentor Session - SEC401	Edmonton, AB	Sep 06, 2017 - Oct 18, 2017	Mentor
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
Mentor Session - SEC401	Ventura, CA	Sep 11, 2017 - Oct 12, 2017	Mentor
Community SANS Albany SEC401	Albany, NY	Sep 11, 2017 - Sep 16, 2017	Community SANS
Community SANS Columbia SEC401	Columbia, MD	Sep 18, 2017 - Sep 23, 2017	Community SANS
Community SANS Dallas SEC401	Dallas, TX	Sep 18, 2017 - Sep 23, 2017	Community SANS
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Copenhagen 2017	Copenhagen, Denmark	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Boise SEC401	Boise, ID	Sep 25, 2017 - Sep 30, 2017	Community SANS
Baltimore Fall 2017 - SEC401: Security Essentials Bootcamp Style	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
Community SANS New York SEC401	New York, NY	Sep 25, 2017 - Sep 30, 2017	Community SANS
Rocky Mountain Fall 2017	Denver, CO	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Sacramento SEC401	Sacramento, CA	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS DFIR Prague 2017	Prague, Czech Republic	Oct 02, 2017 - Oct 08, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Oct 02, 2017 - Oct 07, 2017	Community SANS
Mentor Session - SEC401	Arlington, VA	Oct 04, 2017 - Nov 15, 2017	Mentor