Interested in learning more about cyber security training?

SANS Institute
InfoSec Reading Room

This paper is from the SANS Institute Reading Room site. Reposting is not permitted without express written permission.

Sun Ray Thin-Client and Smart Cards: An Old Concept With New Muscle

The purpose of this paper is to discuss security in thin-client technology with a focus on Sun Ray(TM) thin-client systems from Sun Microsystems. Sun Ray(TM) thin-client differentiates itself from other systems by incorporating the use of a smart card as an additional authentication token to a conventional login and password. In the following sections, a brief introduction to the history and inherent security benefits of thin-clients and smart cards is presented. Next, the paper describes Sun Ray(TM) thinclient's hardw...

Copyright SANS Institute
Author Retains Full Rights

AD
MobileIron EMM Strategy on the right track? Know your security risks. TAKE THE ASSESSMENT

Sun Ray™ Thin-Client and Smart Cards: An Old Concept With New Muscle

Surachet Tanwongsval
GSEC Practical v1.3
April 5, 2002

1. Abstract

The purpose of this paper is to discuss security in thin-client technology with a focus on Sun Ray™ thin-client systems from Sun Microsystems. Sun Ray™ thin-client differentiates itself from other systems by incorporating the use of a smart card as an additional authentication token to a conventional login and password. In the following sections, a brief introduction to the history and inherent security benefits of thin-clients and smart cards is presented. Next, the paper describes Sun Ray™ thin-client's hardware and software architectures and lists businesses that will benefit from implementing the Sun Ray™ system. Finally, the author identifies and presents Sun Ray™ system's security, interoperability, scalability, and performance concerns along with the respective solutions.

2. Introduction

As new viruses and vulnerabilities are discovered daily, the need for securing the enterprise network is more crucial than ever. This, coupled with the constant expansion and increase in networks and PCs, has made this a daunting task for every system administrator. All it requires to compromise a network is, for example, a non-vigilant user to open a virus-infected email attachment further spreading the virus in the organization or forgetting to install a critical update leaving the PC vulnerable to attacks. A solution to prevent such disasters might lie in an old technology of dumb terminals or now known as thin-client. Thin-client technology enables system administrators to centrally manage data, application, resources, and updates. Centralization will come at a price to the users in terms of limited level of customization. However, organizations may be willing to sacrifice any inconvenience for an increase in network security and manageability. The cost benefit of thin-clients also extends to other areas such as energy saving and ease in backing up essential data. Despite all the benefits, there remains a security concern for most thin-client technology. Many, if not all, use a conventional login and password combination for authentication, thus, exposing the system to password security issues.

The introduction of tamper-resistant microprocessors embedded in plastic cards, or commonly known as smart cards, can provide an extra level of security to the current authentication scheme. To gain access to the thin client terminal (Sun Ray™ Appliance), the user must not only have an authorized login and password combination (something you know), but also an authorized smart card (something you have). Sun Ray™ thin-client system provides all the benefits of thin-client with an extra security feature introduced by incorporating the smart card into the Sun Ray™ appliance terminals. To fully understand the technology behind Sun Ray™ systems, it is imperative to first be aware of the general thin-client and smart card technologies.

3. Brief History of Thin-Clients

Thin-clients, previously known as dumb terminals, have been in the networking arena

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

for a long time. They were called dumb terminals because little or no processing was done on the client. They simply displayed the output from the server and communicated the user's input via keyboard and/or mouse back to the server. The main system was centralized with all the data and the applications being stored and managed by a single server or a cluster of servers.

The centralized concept was widely embraced by various enterprises during the 70s due to advantages such as manageability, fault tolerance, central administration and security. However, as the cost of PCs decreased in the 80s, decentralizing the system with individual PCs gained popularity. In addition, PCs provided various features that dumb terminals did not, such as a graphical user interface and user customization. Decentralizing, on the other hand, made administering, maintaining, and upgrading the system an arduous task, as they must be performed locally.

From the late 80s to the mid 90s, a hybrid of both systems known as client/server dominated the networking scene. The server usually handled database management while the PC client handled the applications and the user interface. Important data could easily be backed up and performance was better than sharing a file on a PC. The problem of maintaining, administering, and upgrading (both applications and the OS) on individual PCs remains one of the major drawbacks.

4. Thin-Client Technology

Unlike the client/server model that executes applications locally on the PC, thin-client technology centrally stores and runs the applications on a server, similar to a dumb terminal. This guarantees a uniform application platform across the whole organization. As opposed to dumb terminals' text-based interface, thin-client terminals can support a graphical user interface making the technology more appealing as a PC replacement. Table 1 lists several advantages of the thin-client technology:

Table 1. Advantages of thin-client technology

Security	• System administrator can centrally control, install, and update the necessary applications. This eliminates incompatibility or vulnerability issues that arise from users installing different software versions or forget to install a critical patch. • Since most thin-client terminals do not contain a hard drive, critical data is protected from theft or file corruption.
Financial Benefits	• Thin-client terminals have very few components therefore require less energy than conventional PCs. • Users will spend less time administering their workstation and concentrate more on their job. • Under normal circumstances, only the server's hardware or software needs to be upgraded while the terminals can still be used.
Network Connectivity	• Conventional PCs and laptops require the user to download emails, attachments, and files (e.g. spreadsheet, word processor, presentation slides, and etc.) before they are able to work on them. Downloading takes up local and corporate system and network resources. Furthermore, once the files are modified, they might need to be resent back to the server, requiring even more network and system resources. Thin-client allows remote users to work on the files without downloading the files, as everything is centrally stored on the server.

© SANS Institute 2002, Author retains full rights.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Fault Tolerant	• A thin-client terminal can easily be replaced without compromising user's data, because it is centrally stored. • Important data can easily be backed up and restored if needed.

4.1 What Goes Into Thin-Client Computing

Besides the hardware, which includes thin-client terminals and application and/or data servers, there are other components that make up thin-client computing. First, the system requires an operating system. Windows, Unix, Citrix, and Linux are some of the popular operating environment that runs on the server and supports thin-client.

Second, a sufficient networking infrastructure must be in place. Graphics, audio, and application data are constantly being transferred from the server to the terminals. Although a single terminal running a web browser may require a small amount of bandwidth, hundreds of terminals running graphic intensive applications will require a huge amount of bandwidth. The network design must therefore take the number of terminals, type of applications, and number of sessions into account.

Last, the system requires a robust and efficient back-end centralized application and client management software [7]. Administrators can then use these tools to monitor and analyze the network traffic. Application and hardware problems can therefore be identified and rectified in the early stages before a failure occurs.

4.2 Thin-Client Solutions

Currently, there are several vendors that provide thin-client solutions. Table 2 below gives a brief overview of the various products and their key features. It should be noted that there are three types of Sun Ray™ appliances (1, 100, and 150), but the difference between them is subtle-only in the type and size of the display. This paper will discuss the Sun Ray™ system in general without specifically focusing on any particular type of appliance.

Table 2. Comparison of various thin-client technologies [5]

	Windows 2000 Terminal Services	**MetaFrame XP**	**Tarantella Enterprise**	**Sun Ray™**
Vendor	Microsoft Corp	Citrix System Inc	Tarantella Inc	Sun Microsystem
Platform	Windows 2000 Server and Advanced Server	Windows 2000 application servers	Unix middleware	Solaris on SPARC processor
Supported Application Server Platform	Windows 2000 Server	-Solaris -HP/UX -Windows 2000 -Windows NT	-Windows -Unix -AS/400 -Web based application	-Solaris -Java -Windows NT (via interoperability software)
Supported Clients Platform	-Any Windows client -Internet Explorer with Remote Display Protocol -Unix or Linux workstation (with 3rd party software)	-Widows -Macintosh -Linux -X Windows -Internet Explorer with supported plug-in	-X Windows emulator -Netscape Navigator or Internet Explorer browser with Java applet support	-Sun Ray™ Appliances -X terminal -Unix or Linux workstation with X emulation software

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Security Features	-Encrypted communication between server and clients. -Limit logon attempts and connection time. -Security restriction for individual user or server.	-SSL encryption for data stream between the server and client. -Pass through authentication: centrally stores user's passwords for automatic authentication.	-RSA SecureID and SSL encryption authentication -Encrypted communication between server and client	-SSL encryption for remote administration. -Smart Card authentication and mobility. -Limit logon attempts and timeout option.

5. Smart Card Primer

A smart card is basically a credit card-sized plastic card with one or more embedded integrated circuit chips. This chip is a miniature computer with an operating system, random access memory (RAM) and electrically erasable programmable read-only memory (EEPROM). Remarkably, it has more processing power than the original IBM personal computer.

Smart cards were first introduced as a tool to thwart credit card fraud and counterfeiting by financial institutions in Europe. After the introduction of smart cards, credit card fraud rates have dropped significantly. Smart cards, however, can also provide security and functionality to other industries on a variety of applications from authentication, encryption and security to loyalty programs, health record, and cash replacement. The application seems to be limited only by the processing power of the integrated circuit chip and imagination. This paper presents yet another application of smart cards. Sun Ray™ thin-client technology successfully leverages the advantages smart card authentication with thin-client computing.

6. Sun Ray™ Thin-Client

6.1 Sun Ray™ Architecture

"Simplicity" is the concept of the Sun Ray™ system-literally. The hardware components that make up the Sun Ray™ appliance are just a monitor, keyboard, mouse, graphics card, integrated smart card reader, and network interface card. All the software components are centrally managed, maintained, and updated on a server by a system administrator. Besides a good hardware and software architecture, it is important to note that in order to successfully implement any thin-client architecture, an adequate network infrastructure must be in place. The Sun Ray™ installation guide has recommended the following provisions [16] for interconnectivity to ensure sufficient bandwidth for effective communication between the client and server:

- Sun recommends a conservative ratio of 10:1 (i.e. 100Mbps connection can support 10 clients) for the Sun Ray™ system.
- Avoid using hubs for the Sun Ray™ interconnect network. A hub provides shared bandwidth rather than switched bandwidth.
- Do not configure the Sun Ray™ server as a router as this will place additional burden on the server. Instead, use a dedicated router to route network traffic.
- Switch should meet certain specifications such as using store and forward rather than cut through. Detail specifications can be found in [16].

© SANS Institute 2002, Author retains full rights.

Despite being simple in concept and design, the Sun Ray™ architecture provides a variety of benefits, especially in security and mobility. Some of the features are discussed below:

- **Security:** Two layers of authentication-login/password combination and smart card (token). Secured Socket Layer encryption can be implemented for secured remote administration.
- **Hot Desking:** Sun's terminology for Sun Ray™'s capability to exactly and instantly access the user's session from any appliance on the associated server group. This feature, once implemented with smart card, allows user to remove their smart card, effectively disconnecting the user's session on the appliance while suspending the session on the server, and giving the user the ability to regain their session simply by inserting the smart card into another appliance.
- **Manageability:** Applications and data are centrally managed on the server reducing administrative overhead and increasing security.
- **Interoperability:** Although Sun Ray™ is designed to work on a SPARC server running Solaris 2.6, 2.7 or 8 operating environment, a cross platform application server can easily support other operating platforms such as Windows.

Figure 1. Sun Ray™ 1 Appliance [17]

6.2 Software Architecture [15]

The management of Sun Ray™ appliances can easily be done by using the Sun Ray™ Server Software. The software hosts several functions, which allow efficient system management and administration, user authentication scheme, smart card session mobility, server group management, failover and load balancing. These functions are:

Authentication Manger: The authentication manager is responsible for enforcing the chosen authentication policies to the Sun Ray™ appliances. Every time a user tries to access the system by authenticating using a token (smart card), the Sun Ray™ appliance will query the authentication manager. If the token unique ID is recognized in the database, the authentication manager has to determine whether the user has any live session or not. The session manager provides this information. If an existing session exists, the user will be redirected to the session. Otherwise, a new session will be loaded.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

There are two types of authentication policy:

- *Zero Administration Authentication policy*. This is the default policy that is in effect when the Sun Ray™ appliance is first installed. It is not recommended to simply use this default policy because smart card authentication is not enabled. The user's session is either tied to the appliance's MAC address, if smart card is not enabled, or the unique token ID, if smart card is enabled. When the Sun Ray™ appliance is first powered up or when the smart card is first inserted, the user is automatically passed through to a login window. The group manager will determine which server has the least load, if a server group exists, and pass the request to the appropriate server. A new common desktop environment (CDE) session is then started on the server and the user is prompted for a username and password.

- *Registered Authentication policy*. This policy provides a higher level of security since every token must first be registered before being given access to the system. The registration can either be distributed or centralized. Distributed registration is used in a trusted environment because the users accessing the system with a new token will be prompted for self-registration. This might not be an appropriate solution for public terminals such as an Internet kiosk. Centralized administration provides a higher level of security since the token must first be authorized before the user can use it to access the system.

Session Manager: As the name indicates, the session manager manages the user's sessions. It signals the services to allow all input/output processing to the appropriate Sun Ray™ appliance when a user reconnects or starts a new session. Likewise, the session manager sends a signal to stop all services' input/output processing when the user disconnects. Both the authentication and session manager must be running at all times for the Sun Ray™ system to operate.

Group Manager: The group manager manages load distribution and handling failover. Group manager runs on each Sun Ray™ server in a failover group, if configured. It keeps track of the network topology and group membership as well as processing load on each server. The information will help the group manager to properly response when a user connects.

In the case of load distribution, the group manager will query other servers to see if an existing session exists. If so, the group manager passes on the information to the respective servers. Otherwise, the group manager uses the other server's information on load and capacity to determine which server will receive new sessions. Once the decision has been made, the token is passed on to that server for re-authentication.

When a server does not respond to a keep-alive message sent by the group manager via UDP port 7007, the server is considered down and the information of the server is removed from the network topology. Sun Ray™ appliances that are connected to the failed server detect the lost connection and request re-authentication. The available

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

server's authentication manager intercepts the request. The user's session is then restarted on a new server. When the failed server reconnects and responds to a keep-alive message, the group manager reinstates the server information into the network topology and redirects new sessions to it. However, previously disconnected sessions will not be restored to that server.

Virtual Device Drivers: The virtual device drivers are responsible for translating high level rendering Application Programming Interface (API) to a native Sun Ray™ protocol. All processing and rendering are processed in the server and the pixels are sent to the appliance to be displayed via the virtual device drivers. Similarly, the user's input through the appliance is sent back to the virtual device drivers in the server to be translated for the application. In essence, the Sun Ray™ server, appliance and virtual device driver are analogous to an operating system, hardware device, and device driver, respectively. The driver allows the device and the operating system to communicate.

Peripheral Device Support: Sun Ray™ appliance can access remote or local peripherals through the network or appliance's USB port, respectively. Both methods require the device driver to reside on the server. A remote device manager handles the connections to the remote device and associates the device driver on the server to the peripherals locally attached to the USB port.

Administration Tools: Administration of the policy, users, smart card, server group, and applications are done through the administration tool. The tool is located on the Sun Ray™ server and can either be locally or remotely accessed through a browser based graphical user or command line interface. The communication is secured using SSL, if SSL was setup during the installation.

7. Installing and Administering Sun Ray™ Appliance

Sun Ray™ software version 1.3 is available for purchase in a CD or through FTP. The installation requires a SPARC processor with Solaris 2.6, 2.7 or 8 operating environment on the designated server. In addition, the network interconnect should be correctly provisioned as outlined in section 6.1.

An install script is provided to facilitate the installation of required patches and software. After installing the required patches and software, the network settings of Sun Ray™ server must be configured. Information such as the hostname, IP address, and the network mask of server, first and last IP address of Sun Ray™ appliance, administrative password, enabling of SSL and Sun Ray™ admin server port are required during the configuration. If SSL is enabled, the root certificate authority user and password must be set up. The certificate authority can either be installed on the same Sun Ray™ server or on another host. The author recommends enabling SSL to secure the communication during the administering of the Sun Ray™ server.

Other than the previously discussed features, Sun Ray™ software includes many other utilities and tools that are optional and can be installed when needed. These are the failover group, Sun Ray™ management center, XINERAMA, and controlled

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

browser. More information on the can be found in the Advanced Sun Ray™ Server Administration Guide [14].

The administrative tool is a browser-based GUI that can be accessed by simply using the URL http://hostname:port or https://hostname:port (if SSL is configured), where hostname is the hostname of the Sun Ray™ server and the port is the administrative port (port 1660 is the default). There are various functions available from the administrative tool. More information can be found in the administrative guide. This paper will only concentrate on the system's security and smart card functions.

8. Who Should Use Sun Ray™?

After discussing Sun Ray™ system architectures, one would likely ask what businesses would benefit from deploying Sun Ray™ system? Answering the following questions may help identify whether an organization should consider implementing Sun Ray™ system or not:

- Do you want an additional authentication token (i.e. smart cards)?
- Do you want to secure the network from unauthorized usage but allow authorized users to login via any computer?
- Is there potential for an outbreak of a virus because an employee unknowingly opens an infected file from external sources (diskette or CD)?
- Do you require a regular backup of important data?
- Do you have problems with PCs having nonstandard applications or platforms?

If you answered yes to any of the above questions, Sun Ray™ system may help you resolve the problems. In general, the businesses that will benefit from Sun Ray™ system are, but not limited to, helpdesks, call centers, libraries, banks, university campuses, public schools, and hospitals. Users and system administrators will enjoy all the benefits of thin-client computing and smart cards. Furthermore, smart cards can double as identification cards and can also be used to control physical access into buildings and rooms.

9. Sun Ray™: Concerns and Solutions

Sun Ray™ is a secure, robust, and an efficient system for most applications. However, there are several security, interoperability, scalability, and performance concerns that must be discussed. This section will identify and address the concerns and appropriate solutions.

When a smart card is inserted into a Sun Ray™ appliance, it is powered up by the card reader, runs a power-on-reset, and sends an answer to reset (ATR) to the card reader [6]. The ATR contains specific information on the communication protocol between the card reader and the card. Next, the Sun Ray™ appliance will query the smart card for a unique ID, which is a portion of a reply from a "get data" application protocol data unit (APDU) command. The ID contains unique information such as the smart card manufacturer, smart card chip manufacturer, chip type, batch number, etc that identifies a particular card from other cards. Sun Ray™ system uses this

© SANS Institute 2002, Author retains full rights.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

unique ID to authorize users to the system as well as to tie a user’s session to the token, enabling hot desking with a smart card. There are several concerns regarding security and interoperability with how Sun Ray™ uses this technique. Another concern is the Sun Ray™ system’s scalability and performance when dealing with large networks or graphics intensive applications. The concerns and their respective solutions are presented below:

9.1 Security

Concerns: First, when a user removes his/her smart card instead of logging out, the applications remain active on the server. When the smart card is inserted back, the user will be redirected to his/her session without reauthorization. This hot desking concept is convenient but creates a big security risk. If the card is stolen while a session is active, the unauthorized individual can immediately gain access to the authorized user’s active session.

Second, the unique ID does not have any correspondence to the user’s login ID. This means that a user has the ability to not only login to his account, but any other Solaris account with the same token. Also, the login and password are not locally stored on the smart card. Instead, the login and password are stored on the Sun Ray™ server as a conventional Solaris *shadow* file. This does not take advantage of the security features provided by the smart card. Security is therefore only as good as the Solaris operating environment can provide.

Third, the only encryption available is SSL encryption for remote administration. The system does not provide any encryption option for the communication between the appliance and the server. As a result, the token ID, login, and password are sent to the server in clear text. Using sniffer software such as Ethereal with a hub will allow a hacker to easily view all the packets and extract confidential information.

Solutions: A challenge/response security policy should be implemented for hot desking. This feature will considerably increase the security level because an authorized token, without the appropriate login/password combination, will not be able to gain access to any active session. The drawback is that this process makes the system more intrusive to the user. Another solution would be to set the option to disconnect the user’s session whenever the smart card is removed. The user must then re-authenticate and start a new session when the smart card is inserted.

To fully utilize the security features of a smart card, the card’s unique ID should not be used as an authentication token. Instead, it should be replaced by the following smart card authentication scheme using public key infrastructure (PKI) and digital certificate. In a nutshell, this process will involve storing the user’s login and password, public and private key, and digital certificate in the smart card, requesting the user for password when the card is inserted into the appliance, and authenticating the password by the smart card. Upon successful authentication, a signal will be sent from the card to the card reader, which relays it to the operating system and finally to the certificate authority (CA). The CA will then open a secured communication channel and query the smart card for the digital certificate. Upon verification of the

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

digital certificate by the CA, access is grant to the user. This method should eliminate security risks in storing the passwords centrally on the server.

The use of PKI technology in Sun Ray™ system should also eliminate the need for login/password encryption because the authentication process is performed locally on the smart card and the information is not send across the network. Furthermore, PKI infrastructure allows users to selectively encrypt confidential information using the recipient's public key. This requires much less bandwidth when compared to encrypting the entire connection. Finally, storing the user's information on the smart card has the benefit of inter-network mobility. A user should be able to use his/her smart card to login to, for example, an enterprise's network in different countries as long as the local CA has the information on the other CA that issued the certificate. Detail information on the building blocks of PKI technology and its various applications are available in references.

9.2 Smart Card Interoperability

Concerns: A configuration file provided by the smart card manufacturer is used to control the communication between the Sun Ray™ appliance and a smart card. The configuration file is simply a script written in smart card programming language that queries a unique ID from the smart card using the appropriate APDU commands. Since different types of smart cards from different manufacturers usually have different APDU command, the Sun Ray™ administration software inevitably requires several configuration files to support various types of smart card. The lack of generic configuration file will be problematic when an organization uses a new type of smart card or an unsupported smart card.

Solutions: There are a few standard protocols such as JavaCard API, VisaOP, and EMV that if the smart card architecture and operating system support, any communications done in those standards can read any card supporting them. However, to actually read physical data from a card requires knowledge of the specific APDU commands required for that card. Since each card's set of APDU commands for the most part is unique, the ATR is used to help identify and verify which card is present, and therefore help determine which set of APDU commands will be needed.

Sun Ray™ uses a similar technique in its configuration files to read the unique token ID off from the smart card. It is possible that a combination of various APDU commands can be incorporated into a single configuration file but this file will probably not work for a new type of smart card. The solution in providing support for a new type of smart card is in customizing the existing configuration files with the appropriate ATR and APDU.

It is evident that until a common standard protocol for reading/writing information physically on smart card exists, there must be a configuration file for each specific type of card. Thus, smart card interoperability is not just a Sun Ray™ system issue but rather an issue for smart cards in general. It is therefore advisable for an organization embracing the Sun Ray™ system to consider using a supported smart

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

card or a single type of unsupported smart card to minimize customization.

9.3 Scalability and Performance

Concerns: The performance of the Sun Ray™ system depends largely on the type and state of the interconnect network. If a hub is used instead of a switch or if a switch fails, there could be great repercussion on the performance and/or availability of the system. This presents a challenge for a large network where tens and hundreds of switches or routers are installed. Another scalability issue is the type of application being used in the system. Sun Ray™ performs relatively well for conventional applications such as word processor, spreadsheets, browser, and etc. However, the performance may be questionable for graphic intensive or huge database applications.

Solutions: In a networking environment, it is logical to have the least number of connection points to avoid introducing unnecessary points of failure. Manageability and security of the network also increase if there's a centralized location for all the networking equipment. Therefore, not only should the Sun Ray™ system administrator ensure that the interconnect network is properly provisioned as outlined in section 6.1; he/she should try to have the least number of connections from the server to the appliance. This will enhance performance and facilitate troubleshooting when network connection problems arise.

If a group of users in the organization requires graphic intensive or large database applications, efforts should be made to have a server dedicated to the particular group. However, for an organization that requires a major portion or all of the users to use these applications as standard applications, the option may be to increase the capacity of the Sun Ray™ servers. For these cases, cost might become the organization's major factor in deciding whether a Sun Ray™ thin-client system should be implemented or not because the initial investment could be substantial.

10. Conclusion

Many people may have not realized the various benefits that thin-client technology can offer. They are either unaware of the differences between the current thin-client technology and the text-based dump terminals in the 1970s or the enhanced security, manageability, and performance that thin-clients offer. Currently, there are several thin-client solutions that offer centralized application and data deployment under various platforms from Windows to Unix.

Among the various solutions, Sun Ray™ stands out as the thin-client solution that integrates smart card as an additional authentication token to login and password. However, there are still various security, interoperability, scalability, and performance concerns that should be addressed and resolved.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

11. References:

1. Brown, Jerry. "PKI and Information Security Awareness: Opportunity and Obligation." GIAC Security Essentials. Jan. 24, 2002
URL: http://rr.sans.org/encryption/opportunity.php

2. Cyclops Tech Co., Ltd. "Thin Client History."
URL: http://www.cyclopstech.com.hk/thinhist.htm

3. Fenger, Carl. "The Advantages of Thin Client Computing." adtcom Network Computing. Oct. 22, 2001
URL: http://www.thinplanet.com/tech/generic.asp?f=TDnumber&k=s&v=TD19034

4. Hammond, Eric. "Sun Rays shine bright." FCW Government Technology Group. Aug. 14, 2000
URL: http://www.fcw.com/fcw/articles/2000/0814/tec-sunray-08-14-00.asp

5. Kay, Russel. "Thin-Client Software." Computer World. Feb. 01, 2002,
URL: http://www.computerworld.com/cwi/story/0,1199,NAV47_STO67981,00.html

6. Maxking Tunisia. "Maxking Tunisia Smart Card Info."
URL: http://www.maxkingtunisia.com/smartcardbackground.htm

7. Maxspeed Corp. "Common Sense About Thin Clients." Jan. 2, 2002
URL: http://www.thinplanet.com/tech/generic.asp?f=TDnumber&k=s&v=TD34202

8. Metz, Cade. "Sun Ray 1." PC Magazine. Mar. 17, 2000
URL: http://www.zdnet.com/products/stories/reviews/0,4161,2455865,00.html

9. Seltzer, Larry. "Thin-Client Technology." ExtremeTech. Jan. 15, 2002
URL: http://www.pcmag.com/article/0,2997,s=1579&a=19876,00.asp

10. Starling, Andrew. "Sun Ray 1." Web Developer's Journal. Dec. 14, 1999
URL: http://www.webdevelopersjournal.com/hardware/sun_ray.html

11. Storch, David. "Technology Brief: Public Key Infrastructure (PKI): A Primer." Schlumberger Network Solutions
URL: http://www.slb.com/Hub/Docs/tt/nws/more/tech_briefs/pdf/PblcKey.pdf

12. Storch, David. "Technology Brief: Smart Cards and Public Key Infrastructure." Schlumberger Network Solutions
URL: http://www.slb.com/Hub/Docs/tt/nws/more/tech_briefs/pdf/SmrtCrd.pdf

13. Sun Microsystems. "Sun Ray™ Server Software 1.3 Administration Guide."
URL: http://www.sun.com/products/sunray/docs/1.3AdminGuide.pdf

14. Sun Microsystems. "Sun Ray™ Server Software 1.3 Advanced Administration Guide."

© SANS Institute 2002, Author retains full rights.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

URL: http://www.sun.com/products/sunray/docs/1.3AdvAdminGuide.pdf

15. Sun Microsystems. "Sun Ray™ Server Software 1.3 Detailed View."
URL: http://www.sun.com/products/sunray/software/details.html

16. Sun Microsystems. "Sun Ray™ Server Software 1.3 Installation Guide".
URL: http://www.sun.com/products/sunray/docs/1.3InstallGuide.pdf

17. Sun Microsystems. "Thin Clients Sun Ray™ 1 Appliance."
URL: http://www.sun.com/products/sunray/sunray1/

18. Wilson, Chuck. Get Smart. Mullaney Publishing Group. June 2001

© SANS Institute 2002, Author retains full rights.

© SANS Institute 2002, As part of the Information Security Reading Room. Author retains full rights.

Last Updated: July 20th, 2018

Upcoming SANS Training

Click here to view a list of all SANS Courses

SANS Riyadh July 2018	Riyadh, SA	Jul 28, 2018 - Aug 02, 2018	Live Event
SANS Pittsburgh 2018	Pittsburgh, PAUS	Jul 30, 2018 - Aug 04, 2018	Live Event
Security Operations Summit & Training 2018	New Orleans, LAUS	Jul 30, 2018 - Aug 06, 2018	Live Event
SANS Hyderabad 2018	Hyderabad, IN	Aug 06, 2018 - Aug 11, 2018	Live Event
Security Awareness Summit & Training 2018	Charleston, SCUS	Aug 06, 2018 - Aug 15, 2018	Live Event
SANS Boston Summer 2018	Boston, MAUS	Aug 06, 2018 - Aug 11, 2018	Live Event
SANS San Antonio 2018	San Antonio, TXUS	Aug 06, 2018 - Aug 11, 2018	Live Event
SANS August Sydney 2018	Sydney, AU	Aug 06, 2018 - Aug 25, 2018	Live Event
SANS New York City Summer 2018	New York City, NYUS	Aug 13, 2018 - Aug 18, 2018	Live Event
SANS Northern Virginia- Alexandria 2018	Alexandria, VAUS	Aug 13, 2018 - Aug 18, 2018	Live Event
SANS Krakow 2018	Krakow, PL	Aug 20, 2018 - Aug 25, 2018	Live Event
Data Breach Summit & Training 2018	New York City, NYUS	Aug 20, 2018 - Aug 27, 2018	Live Event
SANS Chicago 2018	Chicago, ILUS	Aug 20, 2018 - Aug 25, 2018	Live Event
SANS Prague 2018	Prague, CZ	Aug 20, 2018 - Aug 25, 2018	Live Event
SANS Virginia Beach 2018	Virginia Beach, VAUS	Aug 20, 2018 - Aug 31, 2018	Live Event
SANS San Francisco Summer 2018	San Francisco, CAUS	Aug 26, 2018 - Aug 31, 2018	Live Event
SANS Copenhagen August 2018	Copenhagen, DK	Aug 27, 2018 - Sep 01, 2018	Live Event
SANS SEC504 @ Bangalore 2018	Bangalore, IN	Aug 27, 2018 - Sep 01, 2018	Live Event
SANS Wellington 2018	Wellington, NZ	Sep 03, 2018 - Sep 08, 2018	Live Event
SANS Amsterdam September 2018	Amsterdam, NL	Sep 03, 2018 - Sep 08, 2018	Live Event
SANS Tokyo Autumn 2018	Tokyo, JP	Sep 03, 2018 - Sep 15, 2018	Live Event
SANS Tampa-Clearwater 2018	Tampa, FLUS	Sep 04, 2018 - Sep 09, 2018	Live Event
SANS MGT516 Beta One 2018	Arlington, VAUS	Sep 04, 2018 - Sep 08, 2018	Live Event
Threat Hunting & Incident Response Summit & Training 2018	New Orleans, LAUS	Sep 06, 2018 - Sep 13, 2018	Live Event
SANS Baltimore Fall 2018	Baltimore, MDUS	Sep 08, 2018 - Sep 15, 2018	Live Event
SANS Alaska Summit & Training 2018	Anchorage, AKUS	Sep 10, 2018 - Sep 15, 2018	Live Event
SANS Munich September 2018	Munich, DE	Sep 16, 2018 - Sep 22, 2018	Live Event
SANS London September 2018	London, GB	Sep 17, 2018 - Sep 22, 2018	Live Event
SANS Network Security 2018	Las Vegas, NVUS	Sep 23, 2018 - Sep 30, 2018	Live Event
SANS DFIR Prague Summit & Training 2018	Prague, CZ	Oct 01, 2018 - Oct 07, 2018	Live Event
Oil & Gas Cybersecurity Summit & Training 2018	Houston, TXUS	Oct 01, 2018 - Oct 06, 2018	Live Event
SANS Brussels October 2018	Brussels, BE	Oct 08, 2018 - Oct 13, 2018	Live Event
SANS Pen Test Berlin 2018	OnlineDE	Jul 23, 2018 - Jul 28, 2018	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced