



EverLink[®] Secure Remote Access and Control (SRAC) Server White Paper

Copyright © 2000 by Anyware Technology, Inc. All rights reserved.

June 28, 2001

Recipients cannot redistribute this document to any third parties without written permission from Anyware Technology, Inc. Anyware Technology, Inc. assumes no liability for any damages caused by the implementation of any software or services suggested in this document. The information in this document is subject to change without prior notice from Anyware Technology, Inc.

Table of Content

Abstract	1
EverLink SRAC Overview	2
1. SRAC— Secure Remote Access & Control and Non-Repudiation.....	2
2. Functions of SRAC Server.....	5
Protections of General Network Applications	5
Digital Signatures on Web Application	6
3. Under the Hook of the SRAC Server.....	6
How to Protect Network Applications.....	6
Network Protection Principle.....	7
How to Sign HTML Forms Digitally.....	8
4. SRAC Server’s Structure	9
5. Authentication Authority	9
6. How SRAC Server Authenticates Users.....	10
7. Registration	10
8. Version	10
9. EverLink SRAC Server and Firewall	11
Install on the firewalled gateway	11
Install in the demilitarized zone (DMZ)	12
EverLink SRAC Architecture	13
1. Network Architecture of B/S Model.....	13
2. Operation Sequence of EverLink SRAC Server.....	14
Take Telnet as an Example:.....	14
○	16
3. TCP Stream Protocol Switch	16
Multiple Connections.....	17
Multiple TCP Ports Support.....	17
Disconnection Conditions.....	18
4. Dynamic Applet page	18
5. Signing HTML Forms.....	21
EverLink SRAC Key Technologies.....	23
1. Security of Java.....	23
Java Language.....	23
Java Library	24
Java Support in the Browser	24
Digital Signed Java Applet	24
2. EverLink SRAC's Implementation of PKI.....	25
3. User’s Access Control.....	28
4. End-to-End SSL.....	28
Protected Network	28
Security Level of SSL Channel.....	29
Switching Protocols	29
5. Secure Authentication.....	30

6. Intrusion Lock.....	30
7. Log and Audit	30
Server Logs	30
Exporting Logs To an Audit Database.....	31
8. Hardware Support	31
9. HTML Form Signing and Verification	32
Signing	32
Submitting with POST Method	32
Submitting with GET Method	32
Packing Signed Form.....	33
Detached Signature Format.....	34
Verification	34
Archiving	34
10. Summary of Key Technologies	35
Keeps Current Software, Hardware and Network Environments Unchanged.....	35
Implementation of PKI/SSL	35
Protocol Switching.....	35
Fine Granular Access Control.....	36
Browser/Server Structure.....	36
Supports Various Hardware Devices	37
Digital Signature	37
Multiple Platform Support and Scalability	38
1. Cross-Platform Solution.....	38
2. Scalability	38
References.....	40
Reference 1: Terminologies.....	40
Reference 2: Standards Compliant in EverLink SRAC Server	46

Abstract

EverLink SRAC (EverLink Secure Remote Access & Control) Server is a secure network operating system. It provides the following business values to a corporation:

- Strong 3 factor user authentication (possession of hardware token, knowledge, and 3rd party verification); avoid weak password problems once for all;
- Secure mobile computing for telecommuters (Virtual Private Network); lets employees and contractors access the network resources securely from their homes;
- Secure extranet applications; lets customers and partners access the network resources securely from their offsite locations; the access methods can be web based or through simple file-sharing on the desktop;
- Non-repudiation on web based business transactions; adds digital signatures on your web site HTML forms;
- Granular access control to the VPN and the extranet; centralized authorization for who can access what network services and when;
- Log all the VPN and the extranet activities;
- Enables your VPN and extranet across all computer platforms such as Apple-Talk protocol.

EverLink SRAC Overview

EverLink SRAC is a secure network operating system based on PKI. It enables any organization to extend its core business applications onto the Internet seamlessly and securely. It fully satisfies the requirements of enterprise/organization secure business communications – not only does it offer a secure tunnel to business applications, but it also offers non-repudiation, scalability and granular access control to the business applications.

1. SRAC— Secure Remote Access & Control and Non-Repudiation

Internal Network Protection and Strong Authentication

In traditional Internet communications, a corporation needs to open many application services on the internal network to the Internet, such as Web service, file service, and Telnet service for its mobile workers, branch offices, customers, or partners. Those applications in most cases are protected only by simple usernames and passwords. Because of the weak password problem, it is extremely unsafe to open those applications directly to the Internet. With EverLink SRAC server, simply open the TCP port used by Secure Sockets Layer SSL protocol (port 443, usually), and the company can then offer secure access to such key applications on the Internet. The SRAC Server not only protects the internal network, but also enables certified Internet users to access internal application servers through the SSL channels. By opening a single port to the public network it will simplify administration and cut management costs. At the same time, because the SRAC Server authenticates any TCP connections to this port with digital certificates, it eliminates the weak password problem is eliminated once and for all. The authentication of the digital certificates is based on PKI technology; it will minimize the security threat from the public network.

Access from Anywhere without the High Cost

With EverLink SRAC, users will be able to access the applications securely and extend

their business communications to any place they wish without any modification. This is especially suited for mobile and remote users. A simple connection to the Internet allows them to securely connect to their internal network, and get access to all kinds of application servers. It ensures security, while reducing the costs of communications greatly.

Low Cost Secure Extranet

With EverLink SRAC, a corporation will be able to build its extranet by letting its customers and vendors access certain network applications (not limited to web applications) securely. The corporation can issue digital certificates to the extranet users, then expose part of its network applications (not all of them) to the users. The applications do not need to be modified. No client software needs to be installed in the extranet users' side. Only users with proper digital certificates can access the applications. All data transferred by the application is encrypted on the extranet. All the activities on the extranet are logged.

Accessing a Wide Variety of Applications

EverLink SRAC supports many application layer protocols, such as Telnet, Xwindow, PCAnywhere, terminal emulations, and Microsoft LAN Manager protocols, such as file-sharing and printer-sharing, and Apple-Talk. It also supports Client/Server applications, such as ERP systems, CRM systems, SCM systems, OA systems, database systems, mail systems, etc. End-users of EverLink SRAC Server don't need to install any client software. They can securely connect to the company's intranet through the Internet and run the same applications as they run them in their office. Therefore, system administrators will find that the SRAC Server frees them from the burden of installing and maintaining client software on the end-users' computers. Unlike an IPsec VPN server, when installing SRAC server, there is no need to make any changes to the company's firewall policies.

With the SRAC server, the user's access right is controlled at the per protocol and per server levels, which regulates who can access which server via what protocol in which period of time during the day. All these are accomplished without any change to the end-user's access rights on the application server.

End-to-end security

To prevent data from falling into the wrong hands during its transfer through non-private networks such as the Internet, many technologies based on PKI have been developed. They fall into two main categories: data encryption technology and authentication technology. The encryption technologies used by the EverLink SRAC include symmetric cipher encryption technologies such as DES, Triple-DES, RC2, RC4, and Advanced Encryption Standard (AES). Asymmetric cipher encryption technologies such as RSA also are used for user authentication. The authentication method used by the EverLink SRAC is digital certificates. With a proper digital certificate from a trust certificate authority (CA), a user can establish SSL channels with the SRAC Server. As a result, he can access all the applications in the internal network that he is entitled to. All the data transferred between the user and the applications are encrypted by the SSL channels.

Data Integrity and Non-repudiation

To ensure data integrity and non-repudiation when data transferring through the Internet, all the forms on the EverLink SRAC protected web server can be digitally signed with the user's digital certificate when submitting. This enables trusted, secure and legally enforceable business transactions on the Internet among organizations, customers, and partners. Any web applications can be integrated seamlessly with the SRAC server to enable the digital signature on its HTML forms

Simple Management and International Language Support

EverLink SRAC is a Browser/Server (B/S) model application. The SRAC Server is accessed by its users through the browsers, and it can also be managed by its administrator through a browser. An administrator can securely manage SRAC through a browser from any place. The SRAC Server also authenticates the administrator via his digital certificate. EverLink SRAC Server supports applications of multiple languages, including English, Chinese, etc.

2. Functions of SRAC Server

Protections of General Network Applications

EverLink SRAC implements a PKI infrastructure to authenticate user identities, control access, encrypted data and so on. The examples of network protocols and network applications supported by SRAC include the following:

CVS	X-window
finger	Telnet
HTTP	FTP
ident	Oracle Database
LDAP	DB2 Database
Lotus Notes	Sybase Database
PCAnywhere	SQL Database
POP2	Informix Database
POP3	WebLogic
Remote Login	Microsoft LAN Manager
Shell	RPC
UUCP	Apple-Talk

EverLink SRAC supports the above applications without the need to install any client software on the end-user's computer. All the connections are through a secure proxy applet on the user's browser. EverLink SRAC server makes:

- No change to the current network devices
- No modification to the codes of network applications
- No change to the configurations of application servers
- No replacement of any DLLs in the Operation Systems

Digital Signatures on Web Application

As an additional protection of the HTTP traffic of a web site, the EverLink SRAC can also sign the HTML forms on the web site with the end-user's certificate. The digital signatures on the HTML forms are very important for the B2B web application, especially when a large fund transfer is involved. The digital signatures ensure non-repudiation on the business transactions.

The SRAC Server provides digital signatures with following characteristics:

Minimizes the impact to users; makes digital signings as close as possible to submitting normal HTML forms.

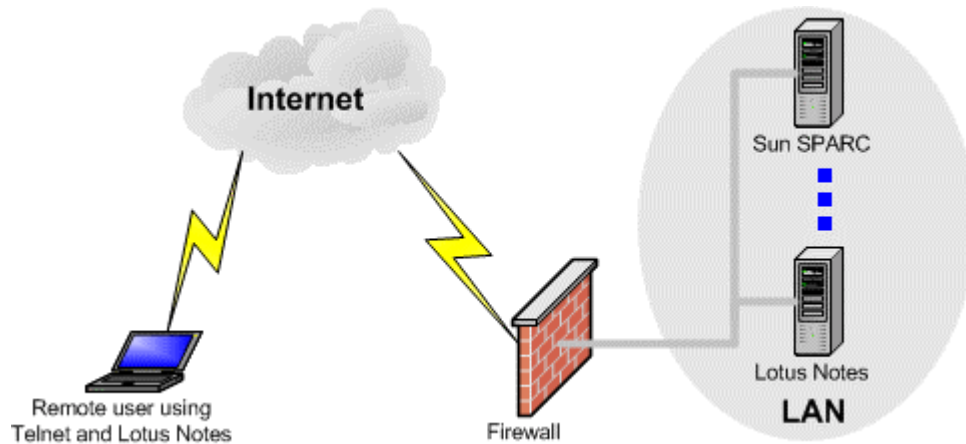
Minimizes the effort to integrate the HTML signing function into a Web Server; makes the Web Server programming as normal as possible.

Supports business workflow through multiple signatures and their verification.

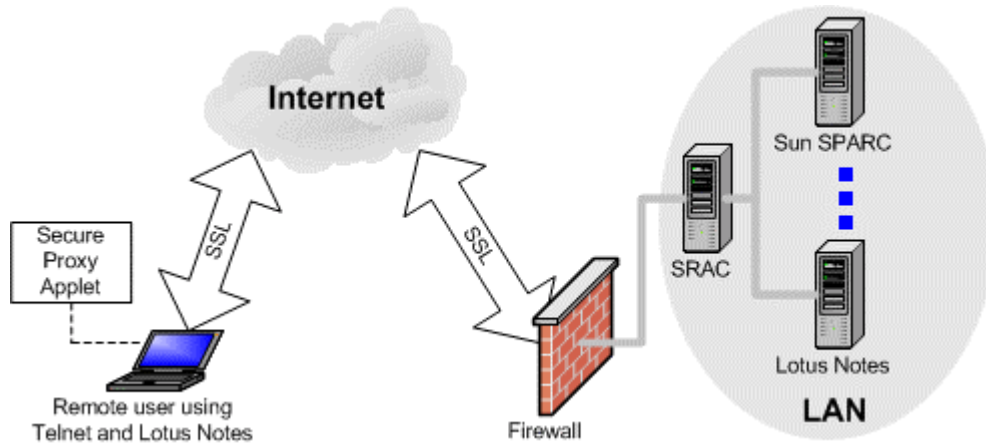
3. Under the Hook of the SRAC Server

How to Protect Network Applications

(1) Traditional Internet application model: Data transmission on the Internet is plain text. Unintended parties easily intercept the data. The network applications are exposed to hacking activities.



(2) EverLink SRAC's application model: It secures the data transferred on the Internet. The data is encrypted with SSL and the end-users are authenticated with their digital certificate.



Network Protection Principle

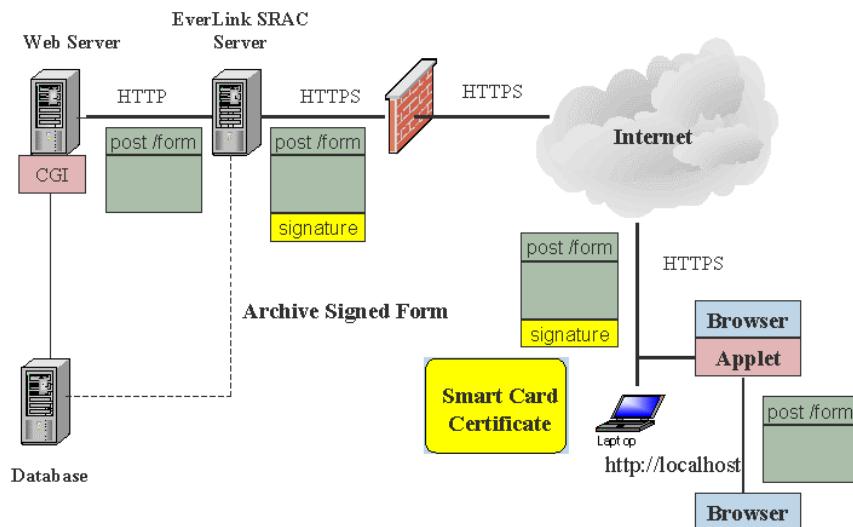
- 1) The SRAC Server protects the application servers. Only one SSL port inside the Intranet, the one on the SRAC Server installed IP address, is opened to the Internet.
- 2) The end-users access the SSL port via HTTPS through their Internet browser, and then download a secure proxy applet after their digital certificates are authenticated by the SRAC Server.
- 3) The secure proxy applet establishes the encryption and certificate-authenticated connection to the remote SRAC Server.

- 4) The application clients (such as the Lotus Notes client, or Telnet) access the remote application through the secure proxy applet in the local browser.
- 5) After the client connects to the applet, the applet will establish an SSL connection to the SRAC Server with the user's digital certificate, then the protocol inside the SSL connection switches from HTTP to the application protocol. The applet switches protocols on the client side and the SRAC server switches protocols on the server side. All the data is transferred through SSL and data in the local machine or protected internal LAN is transferred through application protocols such as Telnet and Lotus Notes.

How to Sign HTML Forms Digitally

The EverLink SRAC Server uses a secure and signing proxy applet to protect a web site and to sign all HTML forms of the web site digitally at same time. It ensures data integrity and non-repudiation when transferring the data on the HTML forms through the Internet. The data is signed with the end-user's digital certificate.

A web server and a database server are on an intranet protected by a firewall. They cannot be accessed directly by the computers outside the firewall. The EverLink SRAC Server is also behind the firewall. The SRAC Server has two network interfaces, one connects to the firewall and another connects to the intranet. The firewall allows the outside computers to access the SRAC Server's private server that binds to a TCP port on the network interface connecting to the firewall. The private server is a certificate authenticated HTTPS server.



After a user has downloaded a secure signing proxy applet, he will access the web site via a URL `http://localhost`. All the browsing on the local URL is as same as browsing the web site directly. When the user submits an HTML form to the web site, the applet will sign the form data and then append the signature to the form data.

After the SRAC Server has received a form data, it will verify the signature first. If a legal user signs the data properly, the signature and the data will be stored into the database. The original form data will be extracted and then sent to the web site.

The digital signatures stored in the database can be audited later. They can be verified independently by third part software.

4. SRAC Server's Structure

The EverLink SRAC Server provides services for end-users and administrators through its Private Server and Administration Server. The administration Server is designed for the administrator to configure and manager the SRAC Server. The Private Server is designed for users to access the SRAC Server's functions. The two servers use two different TCP ports. Therefore, the users and the administrators can securely access data and services from any place.

5. Authentication Authority

To safeguard application servers, EverLink SRAC server controls access by authenticating the users' or administrators' identity using digital certificates. So during the initial setup configuration, the EverLink SRAC administrator needs to select certificate issuers trusted by the EverLink SRAC. The trusted issuers' certificates can be imported directly from EverLink CA Servers, or imported from certification files. The SRAC Server can also import user group information from an EverLink CA server.

6. How SRAC Server Authenticates Users

When users access the EverLink SRAC Server via the HTTPS, the client (web browser) will set up a SSL connection with the SRAC Server. In the SSL handshake process, the SRAC Server will request the browser to send the user's certificate. Upon receiving the user's certificate, the SRAC Server will validate the certificate by checking:

- Whether the issuer of the certificate is trustful;
- Whether the certificate chain is valid;
- Whether the certificate has expired;
- Whether the certificate has been revoked.

After passing the above checking, SRAC will retrieve the user information (such as name, email address, group information, etc.) from the certificate, and match the information with the group information of the SRAC server to decide the user's access rights, such as the accessible protocols, internal server's IP addresses or domain names, etc. If no match is found, the SSL connection will be dropped.

7. Registration

The registration key is required in the EverLink SRAC server setup. The registration key contains three parts:

- Serial number: SRAC server's serial number;
- Session registration key: number of concurrent sessions that SRAC Server will handle.
- Destination registration key: number of servers that the SRAC Server will protect.

8. Version

EverLink SRAC server's current version is 2.00.

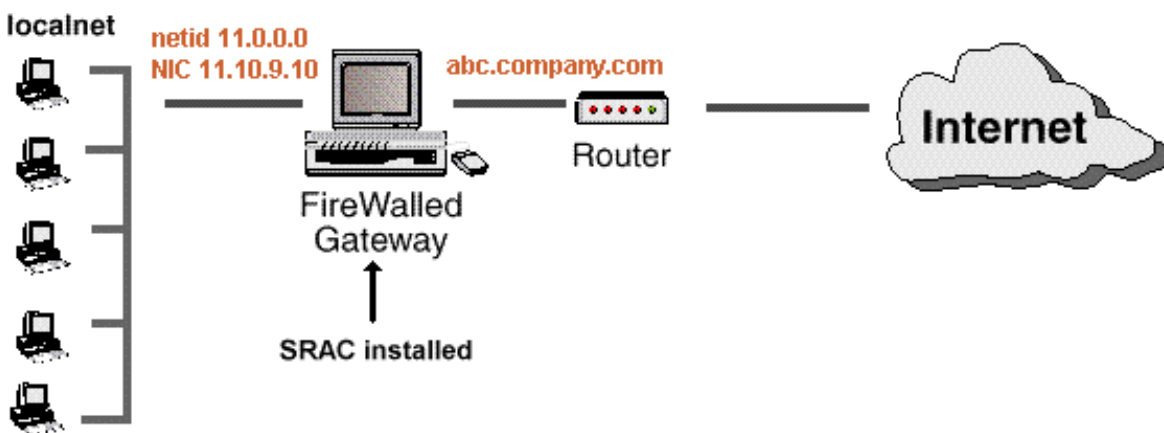
9. EverLink SRAC Server and Firewall

The EverLink SRAC server can work with a company's firewall. The SRAC server must be installed behind the firewall.

There are two ways to install EverLink SRAC Server: install EverLink SRAC and firewall on the same machine, or install EverLink SRAC on a machine in the demilitarized zone (DMZ).

Install on the firewalled gateway

If the firewalled gateway is powerful enough and supports Java Virtual Machine, the EverLink SRAC Server can be installed on it. The figure below is an example of this kind of installation. In this kind of installation, the firewall packet filtering function must allow incoming TCP connections to the SRAC's private server port on the firewall machine's external NIC IP address. The EverLink SRAC Server safeguards these connections. No TCP connections can go through the firewall directly to any other machines inside the firewall. If the SRAC administrator wants to manage the server from a remote location, securely, the firewall packet filtering function must allow TCP connections to the SRAC administration server port on the firewall machine's external NIC IP address. The SRAC server safeguards these connections as well.

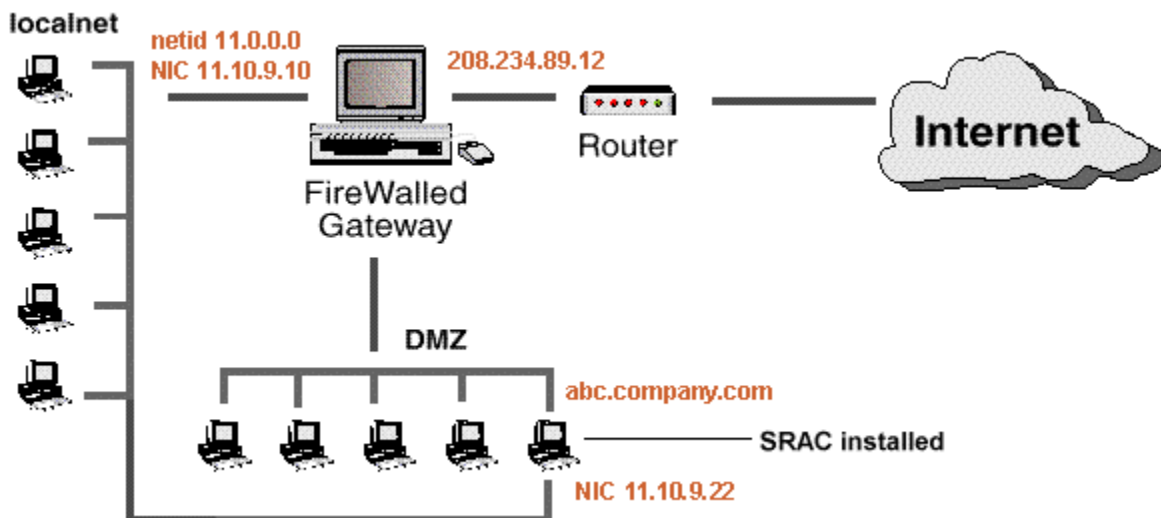


Install in the demilitarized zone (DMZ)

If a company desires to ease the CPU load of the firewalled gateway, or if the firewalled gateway does not support Java Virtual Machine, the SRAC Server can be installed on a machine in the DMZ. The above figure is an example of this kind of installation.

In this installation, the firewall's packet filtering function must allow TCP connections to the EverLink SRAC's private server port on the NIC IP address of the SRAC installed machine. The SRAC server safeguards these connections. No TCP connections can go through the firewall directly to any other machines on the intranet or the DMZ. If the SRAC administrator wants to manage the server from a remote location securely, the firewall packet filtering function must allow TCP connections to the SRAC administration server port on the NIC IP address of the SRAC installed machine. The SRAC server safeguards these connections as well.

In this installation, the firewall's packet filtering must allow TCP connections from the SRAC installed machine to the machines on the intranet. The SRAC server safeguards these connections. These connections must be limited to certain TCP ports used by the users for their applications such as Telnet, Lotus Notes, Microsoft LAN Manager or Apple-Talk.

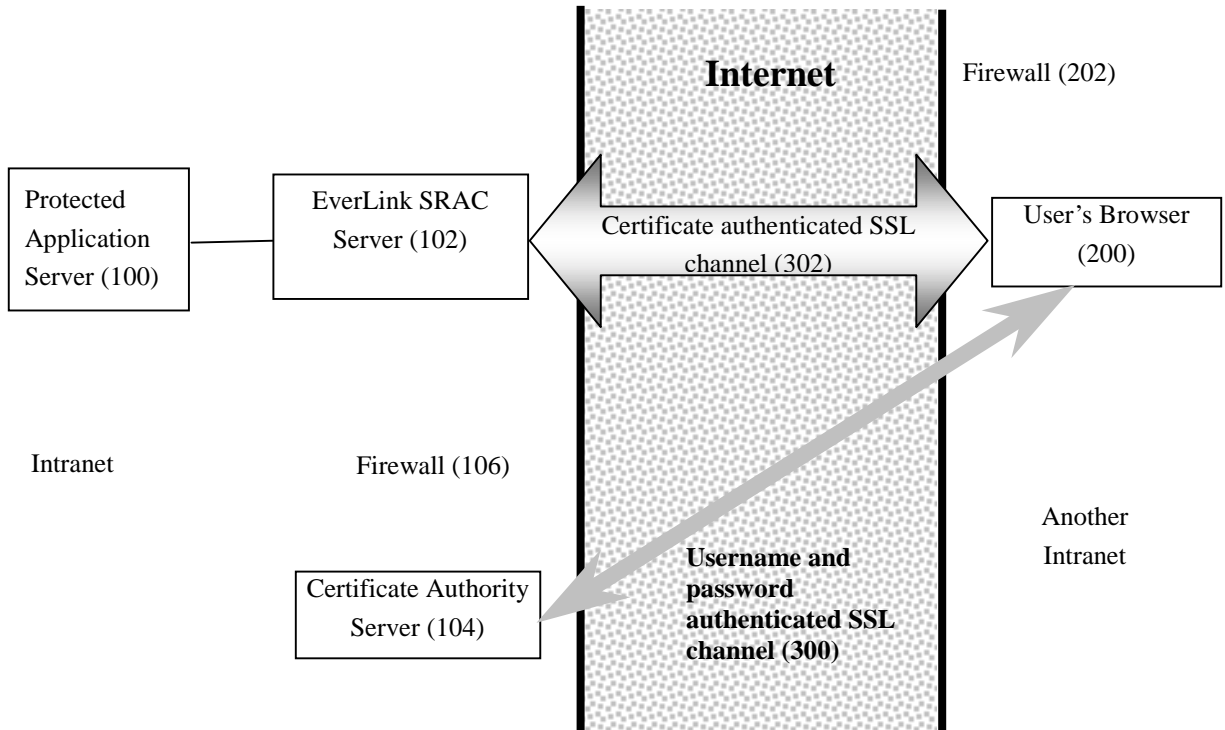


EverLink SRAC Architecture

The EverLink SRAC server is entirely Browser/Server (B/S) architecture. With an Internet browser supporting Java Virtual Machine, users can establish a secure channel with the SRAC Server across the insecure Internet without the need to install any client software.

1. Network Architecture of B/S Model

The following illustrates the network architecture of SRAC.



In the intranet of an enterprise, there is an application server (100) to be accessed from outside the firewall, an EverLink SRAC Server (102) and a certificate authentication

(CA) server (104). All the servers are protected by the firewall (106). The application server (100) is on the same LAN as the EverLink SRAC server (102). The CA server (104) can be installed in another protected network isolated inside the firewall. The CA server (104) can also be a public CA Server outside the firewall.

The EverLink SRAC Server and CA Server each have two servers: Private Server and Administration Server. Private Server and Administration Server are all HTTPS servers. By default, the Private Server uses TCP port 443, and the Administration Server uses port 444, and the firewall opens port 443 on both the CA Server and SRAC Server. The CA Server authenticates its users by user name and password. EverLink SRAC Server authenticates its users by their digital certificates.

Users access the CA Server and the SRAC Server through a browser. The SRAC Server will only accept the certificates issued by the Certificate Servers whose signing certificates are in the SRAC Server's signer certificate database. This means that only users who can obtain their digital certificate from the acceptable CA Servers can get access permission from the SRAC Server.

2. Operation Sequence of EverLink SRAC Server

Take Telnet as an Example:

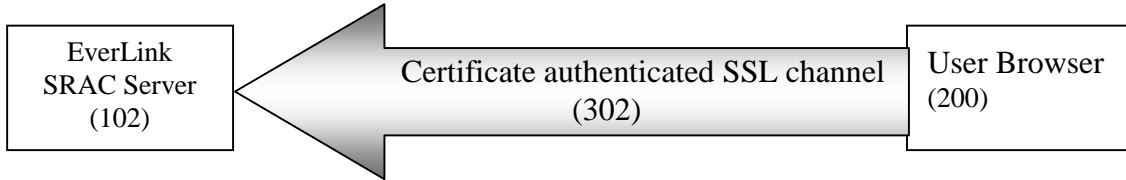
1) Preparation:

To connect to a Telnet server through EverLink SRAC, the user must have a certificate whose issuer is trusted by the SRAC Server. The certificate should include the user's group information defined by organization and organization unit in the certificate.

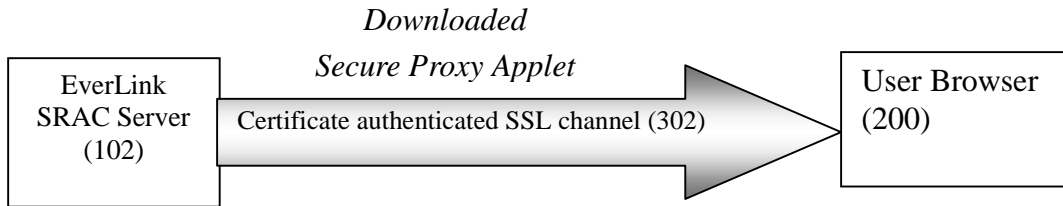
Administrator needs to configure secure channels on the EverLink SRAC Server and assign a group(s) to access each channel. For example, members of the marketing group can access port 88 of Notes server, while members of the developer group can access CVS and access a SUN server through Telnet, etc.

2) After the above step, a user just needs to input the HTTPS URL address of the private server of the SRAC Server to gain access to the application server (100). This address could be the only open port from the Intranet to the Internet, such as <https://srac.anywareusa.com/>. If the SRAC server is connected, the server will request a

user certificate. The browser will then send the certificate to the server. If the SRAC Server trusts the issuer of the certificate, the server will establish a SSL connection with the browser. The SRAC Server (102) will then check the group information in the certificate. If the group member has been granted the right to access a channel/channels, the HTML page including a list of those channels will be sent to the browser. Otherwise the connection will be closed.



3) Click the hyperlink of the Telnet channel and the secure proxy Applet will be downloaded from the SRAC Server.



The Applet will run and listen on the Telnet port of 127.0.0.1 (the default port of Telnet is 23).

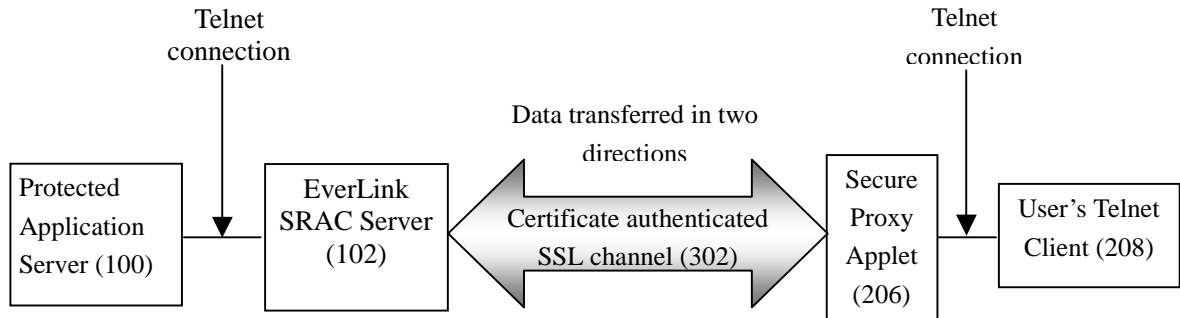


4) Input command: Telnet 127.0.0.1, connects to the Applet. Then, the Applet will set up a secure connection with the EverLink SRAC Server, and send the protocol switch request.

5) The SRAC Server will authenticate the validity of the request. After user's certificate authenticated, the SRAC Server will set up a connection to the Telnet server (100) in the

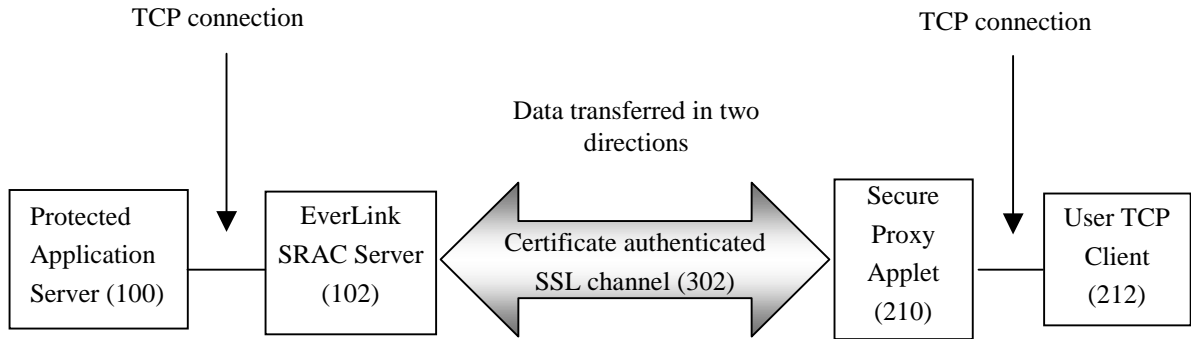
protected network and, at the same time, return protocol switch success information to the Applet.

6) After receiving the protocol switch success information, the Applet will associate the connection to the SRAC Server with the local Telnet connection. Then, the data between the user's Telnet client and Telnet server (100) inside the Intranet can be transferred securely. The user can access the Telnet server securely from any place.



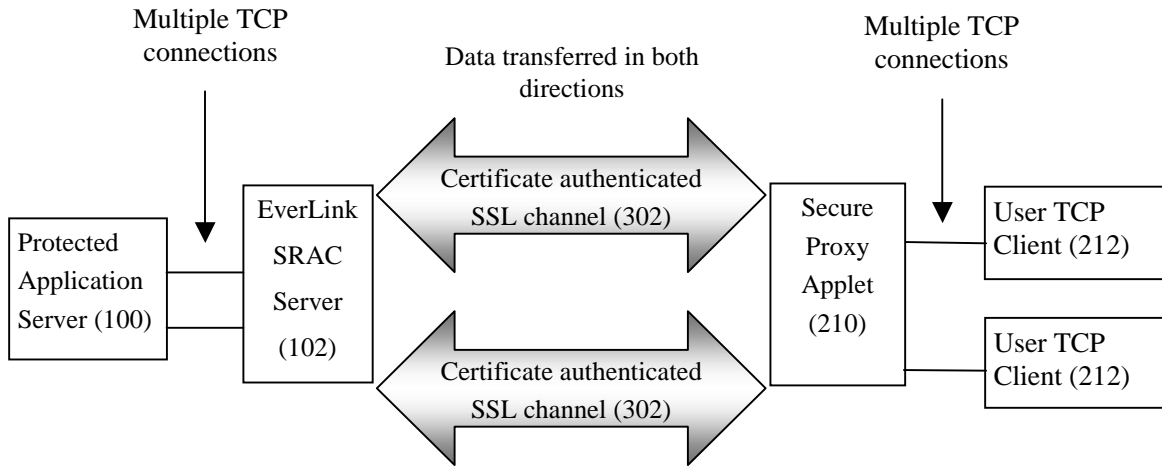
3. TCP Stream Protocol Switch

The principle of the protocol switch is the same for all TCP protocols. The SRAC Server will provide a TCP secure proxy Applet (210) according to the user's certificate and information after the user connects to the SRAC Server. Once downloaded, the Applet will run in the user's browser, and listen on the TCP port of the local host. Then, the user's TCP client (212) will connect to the Applet (210). After that, the Applet will connect to the EverLink SRAC Server (102) securely, and send the protocol switch command to the SRAC Server. The SRAC Server will switch HTTP into the general stream protocol in the SSL channel, and then connect to the TCP server on the application server (100). The data from the TCP client (212) will be securely forwarded to the Applet (210), then to the SRAC Server (102), and finally to the TCP server of the application server (100). The reply of the TCP server will also be transferred from the SRAC Server to the Applet agent, and then to the TCP client.



Multiple Connections

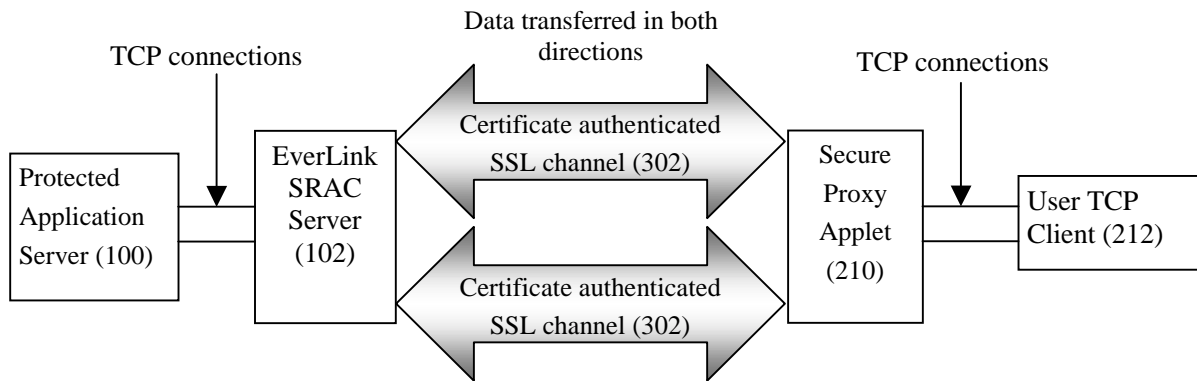
To open the second TCP connection from the client to the server, the user just needs to open the second TCP session from a local Applet (210). There will be one secure proxy Applet instance and multiple protocol client instances on the user's machine. The SRAC Server (102) will also open multiple TCP connections to the protocol server (100).



Multiple TCP Ports Support

The secure proxy applet can be configured to support multiple TCP ports protocols, such as PC Anywhere, etc. After downloading, the Applet will run in the user's browser, and keep listening on the TCP port of the local host, such as, PC Anywhere. Then, the user's TCP client (212) will connect to the port of this protocol. The Applet will then connect to

the EverLink SRAC Server (102) securely, and send the protocol switch command to the SRAC Server. The SRAC Server will switch HTTP into the general stream protocol in the https channel, and then connect to the TCP port in the application server. The procedure of the second TCP connection is the same. The data from the TCP client (212) will be securely forwarded to the Applet, then to the SRAC Server (102), and finally to the TCP server of the application server (100). The reply of the TCP server will also be sent from the SRAC Server to the Applet agent, and then to the TCP client.



Disconnection Conditions

When the following happens, the connection from the TCP client to the TCP server will be terminated:

- Server kills the TCP process.
- User’s certificate has expired.
- The user switches the browser to another URL.
- The TCP protocol client closes the connection.

4. Dynamic Applet page

The following is a sample page including the secure proxy applet in Microsoft IE:

```
<html>

<head>
<meta HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<title>Secure HTML Testing Page</title>
</head>

<body>

<p>In the browser supporting Java, com.anyware.proxy.applets.MSProxyApplet<br> will be
displayed.
<applet CODEBASE="." CODE="com.anyware.proxy.applets.MSProxyApplet.class"
NAME="SecureProxy" WIDTH="380" HEIGHT="270" HSPACE="0" VSPACE="0"
ALIGN="middle">
  <param name=cabase VALUE=mssecureproxy.cab>
  <param name="Remote Server" value="sun.anywaretechnology.com">
  <param name="Remote Port" value="23">
  <param name="Remote Port1" value="25">
  ...
  <param name="Remote Port" value="1025">
</applet>
</p>
</body>
</html>
```

The sample HTML page, including the secure agent Applet in Netscape Navigator, is as follows:

```
<html>

<head>
<meta HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=iso-8859-1">
<title>secure agent HTML testing page</title>
</head>

<body>

<p>In the browser supporting Java, com.anyware.proxy.applets.NSProxyApplet <br> will be
displayed.
<applet CODEBASE= "/" CODE="com.anyware.proxy.applets.NSProxyApplet.class"
ARCHIVE="nssecureproxy.jar" NAME="SecureProxy" WIDTH="380" HEIGHT="270" HSPACE="0"
VSPACE="0"
ALIGN="middle">
  <param name="Remote Server" value="sun.anywaretechnology.com">
  <param name="Remote Port" value="23">
  <param name="Remote Port1" value="25">
  ...
  <param name="Remote Port" value="1025">
</applet>
</p>
</body>
</html>
```

If There is a security problem in the static applet page. Users who have the access rights to parts of the servers behind the firewall, but not to all of the servers, can still access the applet pages that they haven't been granted access rights to. So, all the applet pages in the SRAC server are dynamically generated to prevent such a thing from happening.

There is a server object database storing server object in the EverLink SRAC Server. The server object consists of a server domain name and a TCP port. A physical server may have several server objects. Every server object represents a TCP protocol accessing the physical server.

Another database in the EverLink SRAC database is the access control database, which stores access right objects. The access right object contains a set of server objects and a set of user certificate attributes. When the user connects to the SRAC Server securely, the server searches and retrieves the certificate attributes and server objects from the database. Then the SRAC Server dynamically generates applet web pages using the server object, and sends it to the user. This page is the HTML hyperlink to the specified HTML web page. The user can click the URL to download and install the secure agent applet accessing the specified server, using the specified protocol.

5. Signing HTML Forms

The end-users must have the proper digital certificate installed either on a smart card or on a browser.

The outside computer must use the user's digital certificate to establish an HTTPS connection with the SRAC Server. If the certificate was authenticated through a PKI system, the HTTPS connection would be established. Otherwise, no connection would be allowed.

After the connection is established, the user downloads a digital signed Java applet to his browser. The applet would listen on TCP port 80 (HTTP port) on localhost which then launches another instance of the browser connecting to <http://localhost>.

The applet would connect to the SRAC Server with the HTTPS protocol. Then it would instruct the SRAC Server to relay the connection to the Web Server. The user would then browse the localhost as if he were browsing the Web Server. The information transferred between the user's computer and the Web Server is encrypted. The access rights of the Web Server are guarded with the PKI system.

The applet also monitors the HTTP requests sent by the browser. If it found a request that sends a form to the Web Server, it would sign the form with the user's certificate. Then the form and the digital signature would be sent to the SRAC Server together.

After the SRAC Server gets the form and the digital signature, it will try to verify the signature. If the signature cannot be verified, the SRAC Server will respond with an error message. The wrong form will not be sent to the Web Server. If the signature is verified, the original form will be sent to the Web Server. The correct form and its signature will be archived into the database server for future verification and audit. The Web browser can process the form as usual, such as starting a CGI script to process the form data. The SRAC Server would then pass the response from the Web Server back to the user's computer.

EverLink SRAC Key Technologies

The EverLink SRAC is a new generation of browser/server software that meets the challenges of the Internet age. It offers the convenience of global data exchange to users. At the same time, it ensures the external security and the internal security of various data exchanges. Security is the cornerstone of the EverLink SRAC design.

This section explains the basic concepts and the technologies used in the EverLink SRAC.

1. Security of Java

The EverLink SRAC server software is developed in pure Java. Java can be used as an independent application, and can also be downloaded to run in a web browser as an applet. In the latter case, the web browser must support the Java interpreter and runtime library. In other words, the Java applet program can be downloaded by the web browser, and then executed by the Java interpreter of the web browser. In this model, the security of Java is implemented in the following aspects.

Java Language

Java is an object-oriented language. As to the security, the most important feature of Java is its access control technology.

- Control access rights to variables and methods in the object: to prevent the illegal access of variables or methods by a distrusted code.
- Provides a final method to prevent code reuse. If the type of class or method is final, then the class or method can't be inherited or overridden. Thus, this avoids the class or method being overridden or inherited by a malicious program.
- Type checking: When compiling or running, Java checks the validity of the types of variables, to avoid an illegal type switch.
- No pointer type in the data structure.
- Garbage collection: Java replenishes the unused memory automatically, diminishing any hidden security problems.

- Java keeps the encapsulation of the class name space by using package technology: Classes with the same name in different packages will not lead to confusion or security negligence.

Java Library

Because the Java library is part of the runtime type, it provides the access method to the system resource. Therefore, it is very important to use the library correctly. The access rights control to the library is based on the three principles listed below:

- Java provides access control rights to methods and variables.
- Loading of an imported module using the specified ClassLoader.
- Checking the validity of the operation by calling the global class – SecurityManager.

Java Support in the Browser

The web browser plays a very important role in the area of security. The web browser defines and implements a security policy for running downloaded Java programs. The web browser supporting Java includes a Java interpreter, a runtime library, and classes, which implement the SecurityManager and the ClassLoaders.

The SecurityManager controls the access rights to the core system resources, such as the file manager, the network, the process manager, the user environment and the system calls. The security policy of the web browser can be adjusted according to requirements, and complete control can be achieved when necessary.

Digital Signed Java Applet

To authenticate whether or not an applet program is from a reliable software developer, the applet can be signed with the certificate of the developer. The digital signature is based on PKI technology.

- Only the developer who owns the private key can sign the applet.
- Any one who is able to access the public key can also authenticate the digital signature.
- Any modification to the signed applet (even just one bit among a large file) will make the signature illegal. Therefore it makes sure the applet is never modified

(either by accident or intension) since the developer signed it. Thus, the digital signature ensures integrity of a Java applet.

The ClassLoader can verify the signature first, and then load the downloaded program from the signed applet. If the applet's signature has been verified and the signer is considered to be a trustful developer by the user, then the specified right will be granted to the applet, allowing access to system calls and files. The SRAC's security proxy applet is signed and published by Anyware Technology Inc.

2. EverLink SRAC's Implementation of PKI

The EverLink SRAC utilizes PKI technology, and is completely transparent to users. As long as the user knows how to use his certificate, the rest is taken care of by the SRAC Server.

The following are the 4 features in EverLink SRAC's PKI implementation.

- Security: High strength encryption algorithms and various security measures.
- Interoperability: Capability of exchanging information, certificates and services with other software compliant with PKI standard.
- Flexibility: Configuring PKI system to meet the corporate security needs with minimum efforts.
- Ease-of-use: easy management and usage of the digital certificates for both administrators and end-users.

Security

EverLink SRAC Server supports various encryption algorithms, including symmetric algorithms, such as DES, Triple-DES (168 bits), RC2, RC4 and AES (128 bits, 192 bits, 256 bits) etc, and asymmetric algorithms, such as RSA (1024 bits), and hash algorithms such as MD5 and SHA-1.

Interoperability

PKI technology in the EverLink SRAC Server is compliant with the following industry standards, so it's easy to interoperate with other software programs and environments.

Table 1 PKI standards compliant in the EverLink SRAC Server

Standard	Definition	Description
X.509 Version 3	Format and content of digital certificate	Different PKI vendors can't exchange certificates without certificate format standards.
CRL Version 2	Format and content of the Certificate Revocation List (CRL)	Used in the exchange of CRL between CA and program that needs to verify the certificates.
PKCS Serial	Format and procedure of exchange and distribution of Public Key Infrastructure	Enables users to request and revoke certificates, exchange certificates and encrypted messages, etc.
SSL Version 3	Encryption in the network session	SSL is the most well-known and widely used security protocol for data transmission.
S/MIME	Message digital signature format.	S/MIME is the most well-known and widely used security protocol for digital signatures.

Flexibility

The EverLink SRAC Server provides flexibility to meet corporate security needs in the following aspects:

CRL

Due to various reasons, the user's certificate may need to be revoked. For example, the private key has been divulged, the information in the certificate has changed or the certificate will never be used again. There is list of revoked certificate serial numbers along with revocation reasons. The information regarding certificate revocations will remain in the CRL list, until the certificate expires. Then the CA will delete the certificate from the list. During signature verification, an application can check the CRL to make sure that the certificate and the key pair are still valid.

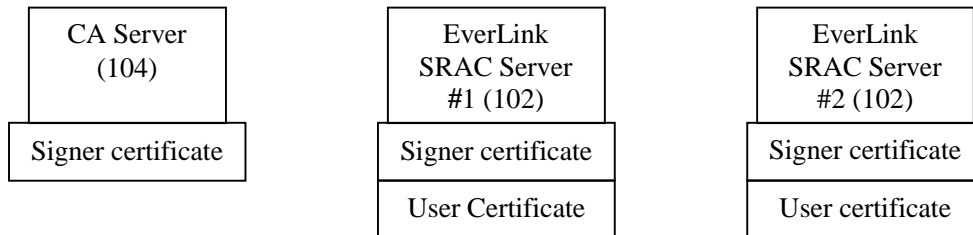
In traditional PKI application software, there is usually no good way to transfer certificates and CRL's between sender and receiver. EverLink SRAC Server can retrieve CRL information from the CA Server online. With frequently updated CRL's, the SRAC Server offers real-time access controls.

Certificate Authentication

When SRAC authenticates users, the process of searching, matching and verifying of certificates is all transparent. Sender and receiver (whatever program or person) need not know how it proceeds and what standards it follows.

During SSL authentication, SRAC only trusts the certificates issued by the specified certificate authority servers. If the certificate isn't issued by the trusted certificate server, the SSL connection will not established. Therefore, with the specified certificate issued by the specified server, the user can access any SRAC Servers that accept CA's of this certificate. The SRAC Server can retrieve the signer certificate securely from the CA server.

The SRAC Server can trust multiple certificate issuers and root certificates. In another aspect, one certificate authority (CA) can work well with multiple EverLink SRAC's. Thus, it can avoid the synchronization hassle of SRAC Server user databases in different locations.



Ease-of-Use

PKI can be easily set up by the administrator of the SRAC Server.

- Certificate management: Administrators can assign which root certificate to trust from the root certificate manager.
- Authentication criteria: When configuring the SRAC Server, the administrator can specify what information in the certificate is used to authenticate himself according to the security requirement. The information such as country, company, and department, etc, can be used as criteria in addition to the name and email address in the certificate.
- Security level setup: Security level can be adjusted through the SRAC setup program, according to the requirements of the corporation. For example, if high security is required in some corporations, then the encryption algorithms of low security level can be deleted.

3. User's Access Control

EverLink SRAC Server implements role-based access control. According to the different departments of a corporation, the administrator can create a hierarchical structure of user groups to determine access rights of the users, even pinpointing access to a single user within a given time period to gain access to a single server's TCP/IP port.

- Access control is assigned by group. A user group can be imported from a CA Server, and controlled centrally.
- User information is stored in the CA Server, eliminating the need for the SRAC Server to synchronize user database with the CA Server.
- System is easy to configure, and a secure channel is down to server IP address and protocol.

4. End-to-End SSL

The EverLink SRAC Server establishes an end-to-end (source-to-destination) SSL channel down to the secure proxy applet downloaded by the user's browser during data exchange, regardless of the positions of the source and destination relative to a firewall. This channel completely secures data exchanges, such as file sharing, e-mail message retrieving and FTP on a TCP/IP network.

SSL is a security protocol that is between transfer and application layers. The symmetric encryption key is generated in every network session, and is based on negotiation between the server and browser. It is generally associated with web applications and HTTP (named as HTTPS) protocol. The SRAC Server uses SSL to secure generic TCP/IP applications.

Both browser and server must have certificates, issued by CA, which is trusted by the two sides, in order to authenticate each other. Once client and server have authenticated each other, they will negotiate a session key and then communicate securely. The identity information in the browser side certificate can be used as criteria for the access control policy.

Protected Network

Protected networks are the TCP/IP networks on your intranet. Usually they are Local Area Networks (LANs) inside your firewall. The SRAC server guards those networks from unauthorized access from the Internet. Only users with an authenticated certificate

can establish an SSL channel with SRAC Server and access resources such as Notes Server, and Web Server on the protected network.

- The EverLink SRAC Server keeps a list of protected networks.
- An item on the list has two values, a subnet netid and a NIC card IP address that has a routing path to the subnet.
- The NIC card must be on the SRAC Server installed server hardware.
- The SRAC Server administrator must enter the list during the SRAC Server initialization.

Security Level of SSL Channel

EverLink SRAC fully supports SSL suites, and offers easy selection of the SSL suites to the administrators during the server setup.

- 1) Multiple encryption levels: SRAC Server supports multiple encryption level, and the administrator can choose the encryption levels to meet his requirement. These encryption levels are: RC2 encryption (128 bits, 64 bits, 56 bits and 40 bits), DES encryption (40 bits and 56 bits), Triple-DES encryption (168 bits) and AES (128 bits, 192 bits, 256 bits).
- 2) Priority control: SRAC Server will search the SSL encryption levels automatically according to the priority of encryption levels set by administrator. And the administrator can add, modify or delete the encryption levels. For example, if the system administrator thinks that it is not secure to encrypt with just 40 bits key, he can delete 40 bits levels from the SRAC setup program. The reason for EverLink SRAC keeping these low level encryption algorithms is to connect with computers using low encryption levels in some countries.

Switching Protocols

EverLink SRAC has the ability to provide transparent secure access and control is due to a key technology – protocol switching. The innovation implements the switching from the HTTP protocol to other protocols. On both the SRAC Server and the secure proxy applet sides, the data of the ordinary application protocols need to switch to HTTP protocol before being transmitted on the Internet, and then data will be transferred over to an SSL channel. The reverse process is also required once the data arrives at the destination.

EverLink SRAC uses the following methods to switch from HTTP to other protocols. The SRAC Server defines a special HTTP GET command for an HTTP Client (a browser

or an applet) to request switching the protocol inside an SSL tunnel. After the authentication check, the SRAC Server returns a "switching protocols" (101 response code) response to the Client. Then the SSL tunnel is established. In this way, any network application can use its custom protocol after the SSL connection is established with the SRAC Server. The data can be transferred with an encrypted format inside a certificate authenticated SSL channel over the insecure network such as the Internet, and then transferred with original application protocol format over the secure network such as the intranet.

With this technology, any network applications can be secured transparently and easily by the EverLink SRAC Server without any modification.

5. Secure Authentication

In the EverLink SRAC, the password or pin number will never be transmitted on the network. The three-factor authentication process is always carried out in the PKI environment.

6. Intrusion Lock

The EverLink SRAC allows three certificate authentication attempts. If all three attempts fail, the EverLink SRAC creates an intrusion lock. This lock will be imposed on the IP address where the failed attempts came from. Further connection from the locked IP address is forbidden.

7. Log and Audit

The detail logs and audit trails are important in any information security system. The EverLink SRAC provides forcing logs, forcing duplications and a JDBC interface to export logs and duplications to an auditing database.

Server Logs

The EverLink SRAC Server has detailed access logs and error logs. The access logs record normal activity on the EverLink SRAC Server. The error log records server errors and any authentication failures and intrusion locks.

Two HTTP servers of an SRAC Server keep detailed logs on the activities and the errors. There are three kinds of logs:

- Administration server errors: Errors record of the SRAC administration server.
- Private server accesses: Access activity record on the SRAC private server.
- Private server errors: Errors record of the SRAC private server.

In the access log, the system records access time to the server, the user name, the domain name and the web request command (For example, GET /internal-cgi/applycertfrombrowser HTTP/1.1) and request object (For example, https://www.my_SRAC.com/title.html) from the browser. The server's response and message when an error occurs are also recorded in addition to the above information. To offer deep level information to the administrator, the log information records all the lower level events. The system therefore, requires the administrator to have certain knowledge about the security technologies of the whole system.

The EverLink SRAC Server keeps the previous seven-day's logs. If the administrator wants more than seven-day's worth of logs, the logs must be backed up to an auditing database.

Exporting Logs To an Audit Database

The EverLink SRAC can export its server logs to an auditing database. The EverLink SRAC provides a JDBC interface to export logs to the database.

The exporting classes must match the individual customer's auditing database. They must be customized to each customer. They are part of the EverLink SRAC application-programming interface (API).

8. Hardware Support

Hardware vendors have developed various encryption cards to meet the demand of the PKI application market. The EverLink SRAC supports a variety of this kind of hardware, which is divided into two categories:

Certificate/private-key storage media: Mainly used in storing a secret key safely for client and server applications, including:

- Electronic keys such as iKey
- Electronic buttons such as iButton
- Smart cards that support Microsoft Crypto API or PKCS11.

Crypto Accelerators: Mostly used in the server applications. Crypto accelerators provide high-speed data encryption and decryption to reduce the CPU bottleneck caused by software encryption, include the following:

- NForce e-commerce server accelerator by nCipher, which also has private key storage and protection functions;
- CryptoSwift e-commerce server accelerator by Rainbow Technologies,
- Other server accelerators that support PKCS #11.

A user can choose whether to use a hardware device or not. For example, a user can choose to store a certificate in the certificate database of a browser (Internet Explorer or Netscape), or in the installed storage devices. The administrator can select and configure accelerators when setting up the SRAC Server.

9. HTML Form Signing and Verification

Signing

When the secure proxy applet monitors the HTTP requests sent by the browser, it will check the first line of a request (HTTP command line). It will check the line against the following patterns:

the line starting with POST (case insensitive);

the line starting with GET (case insensitive) and there is a question mark '?' following the GET such as GET /server/form1?a=b&c=d HTTP/1.1.

Those two patterns indicate the browser sending a form to the Web Server.

Submitting with POST Method

The first pattern indicates the form using POST method. The HTTP request should include a Content-Length header that tells the Web Server how large the form data is. If the applet could not find the Content-Length header, it will not to sign the request. The request will be simply passed to the Web Server by the applet. If the applet found the correct Content-Length, it would buffer the request content then sign the content.

Submitting with GET Method

The second pattern indicates the form using the GET method. The form data is included in the first line of the HTTP request. The data starts at the character following the

question mark and ends at the character before the space character. The applet will extract the form data from the HTTP command line, and then signs them.

The reason that only form data is signed and not the whole HTTP request is that the form data can be reconstructed later to be verified. It is hard to reconstruct the original HTTP request for later verification.

Packing Signed Form

After the applet signing the form data, it will pack the HTTP request and the signature into a special HTTP POST request then sends the repacked request to the SRAC Server to process. The format of the special HTTP POST request is shown as bellows:

```
POST /internal-cgi/signedform HTTP/1.1
Content-Type: multipart/form-data , boundary=AaB03x
Content-Length: 6890
...

--AaB03x
content-disposition: form-data; name="request"
Content-type: application/http-request
Content-Transfer-Encoding: binary

origial HTTP request
--AaB03x
content-disposition: form-data; name="signature"
Content-type: application/pkcs7
Content-Transfer-Encoding: binary

signature
--AaB03x--
```

Here, the boundary is a randomly generated string. The Content-Type is a necessary header. Other type headers are optional. The Content-Length header is strongly recommended.

Detached Signature Format

The signature is in the detached PKCS #7 format. It means that the PKCS #7 data contains only the signature. It contains no signed data itself. The PKCS #7 data contains a signer's certificate, signer information and signature.

Verification

After the SRAC Server receives the special HTTP POST request from the applet, it will extract the original HTTP request and the signature from the request content. Then it will extract the form data in the same way as the applet.

The form data will be used to check against the signature. If the check failed, the SRAC Server would respond with an HTTP error to the remote client directly. The original HTTP request and fake signature will be archived into the database for future audit purposes.

If the form data passed the signature checking, the original HTTP request will be passed to the Web Server. Before the request is sent to the Web Server, the signature identifier would be added to the request.

Archiving

The keys to correct signatures and the fake signature database table are different.

Correct Signatures

The key for the correct signature table will consist of the user's certificate issuer and the certificate serial number hashed with MD5 concatenates time stamp in milliseconds beginning from 1970. The details are described as follows:

```
HEX_STRING(MD5(issuer,serial_number))+":"+HEX_STRING(processed_time_ellipse_from_1970_in_milliseconds)
```

The key is also used as a signature identifier.

The form data and the signature will be archived in its original data format.

Fake Signatures

The fake signature may have no user certificate at all. The key to the fake signature table will consist of the form data hashed with MD5 concatenates the time stamp. The details are described as follows:

```
HEX_STRING(MD5(form_data))+":"+HEX_STRING(processed_time_ellipse_from_1970_in_milliseconds)
```

The key is also used as a fake signature identifier.

Because the original HTTP request contains a lot useful information about the fake signature, the whole HTTP request will be archived into the database.

The request data and the fake signature will be archived in its original data format.

10. Summary of Key Technologies

Keeps Current Software, Hardware and Network Environments

Unchanged

- Protects the current investment and resources.
- Keeps user's habit unchanged.
- Reduces the cost of user training, data conversion and system maintenance caused by adding new security functions.
- Does not compromise the system security.

Implementation of PKI/SSL

- Secures access and control based on digital certificate authentication.
- Ensures data security and integrity.
- Is designed to work seamlessly with EverLink CA Server, and compatible with other CA systems.
- Is fully compatible with all popular Internet browsers.

Protocol Switching

Using the SRAC Server's unique protocol switching technique and an Internet browser, a user can not only access different protocols on different servers on the LANs inside

firewall, but can also access multiple protocols on the server at the same time. The SRAC Server supports a broad range of business applications such as:

Lotus Notes/Domino Server

Relational databases including Oracle and Informix

Telnet server

SMTP server

POP3/IMAP mail server

Web server

Unix/Linux/AIX server

Microsoft LAN Manager

Apple-Talk

Fine Granular Access Control

By supporting the application of a certificate extension field, EverLink SRAC can provide access control with hierarchical structure, and even pinpoint access to a single user within a given time period to gain access to a single server's TCP/IP port.

- Access control is assigned by group. User groups can be imported from EverLink CA Server to implement central management.
- User information is retrieved from the certificate, without need to keep it locally and synchronize the user database.
- System is easy to configure, the addresses and protocols of accessing servers can be centrally controlled by the administrator.

Browser/Server Structure

Entirely based on browser structure, user doesn't have to install any client software.

- Multiple choices: According to requirements, a user can install certificates in browsers, such as IE, Netscape certificate database, or electronic devices such as iKey.
- Easy to use.
- No need to install client software.
- Easy to maintain system.

Supports Various Hardware Devices

- iKey
- iButton
- NFast e-commerce server accelerator by nCipher,
- CryptoSwift e-commerce server accelerator by Rainbow Technologies,
- Other server accelerators that support the PKCS11.

Digital Signature

The SRAC Server provides digital signatures to the HTML forms on the web site that it protects. The digital signatures ensure data integrity and non-repudiation of the web transactions.

Multiple Platform Support and Scalability

1. Cross-Platform Solution

The EverLink SRAC Server is developed entirely in Java. It is a 100% Java application. Therefore, it can run in all the systems supporting the Java virtual machine.

Supports multiple software platforms:

- Windows 98/NT/2000
- Unix/Linux
- Novell Netware
- IBM AIX

Supports multiple hardware platforms

- Intel/AMD serial PC and server
- IBM RS6000
- Sun SPARC

System requirements of server

- CPU: above Pentium 166MHz, RAM: 64M or above, disk space: 30M or above
- All machines support Java virtual machine, including main frame
- Tested successfully in Windows NT, Solaris 2.x, Novell Netware 5.0, AIX and Linux, etc.

2. Scalability

The scalability has two issues; how the SRAC Server handles large amounts of users and how the network handles the encrypted data transfer load.

Due to the EverLink SRAC architecture, the most CPU intensive operations such as RSA key pair generation and digital signing are distributed to the client side. The SRAC server does a part of the encryption operation such as SSL handshaking and symmetric encryption operations. In a real case, a SRAC Server with one hundred users can operate on a 90 MHz Pentium Windows NT server.

In a large user base case, a more powerful server other than Windows NT can be used for the SRAC server. The EverLink SRAC Server is written in pure Java, and any

hardware and operating system which supports Java Virtual Machine can be used. The crypto accelerator hardware can be used on the server side to speed up the SSL handshaking process. The cluster server with server load balancer can be used for a large public ASP site.

References

Reference 1: Terminologies

3DES: An encryption algorithm, like DES, it encrypts with three 64-bit keys. It has many different implementations, but they all encrypt, decrypt, and encrypt the same data block with three different keys.

AES: Advanced Encryption Standard. AES is much stronger, and more effective than 3DES.

CA (Certificate Authority): A trusted third-party organization or company that issues digital certificates used to create digital signatures and public-private key pairs. The role of the CA in this process is to guarantee that the individual granted the unique certificate is, in fact, who he or she claims to be.

Certificate request: A Certificate request is a group of encoded text including the content administration input during CA setup. It's required by most CA's when issuing server certificates.

CVS: CVS is the Control Version System. In UNIX, it is used for recording modification of source code. Version changes and who changes it and what has been changed can all be recorded. A certain version can be retrieved from CVS. It can be used, not only on a single machine, but also by many people working on the same project.

Decryption: The translation of cipher text into plain text.

DES (Data Encryption Standard): is a key encryption method widely used in world. It was developed by IBM in the 70's, and was accepted by the American National Standard Institute (ANSI). In this standard, 72,000,000,000,000,000 (72Q) keys are available. The encryption key is randomly selected from the 72Q keys to each piece of information. Both encryption side and decryption side should use the same key.

Digital certificate: A digital certificate is issued by a certification authority (CA). It

contains your name, a serial number, expiration date, a copy of the certificate holder's public key (used for encryption messages and digital signature), and the digital signature of the certificate-issuing authority so that a recipient can verify that the certificate is real.

DMZ: A Demilitarized Zone is used by a company that wants to host its own Internet services without sacrificing unauthorized access to its private network. The DMZ sits between the Internet and an internal network's line of defense, usually some combination of firewalls and bastion hosts. Typically, the DMZ contains devices accessible to Internet traffic, such as Web (HTTP) servers, FTP servers, SMTP (e-mail) servers and DNS servers.

DNS: Short for Domain Name System (or Service), an Internet service that translates domain names into IP addresses. Because domain names are alphabetic, they're easier to remember. The Internet however, is really based on IP addresses. Every time you use a domain name, a DNS service must translate the name into the corresponding IP address. For example, the domain name www.example.com might translate to 198.105.232.4.

Electronic fingerprint: generated by a set of hash algorithms and used for identification.

Encryption: The translation of data into a secret code. Encryption is the most effective way to achieve data security.

Ethernet: A local-area network (LAN) protocol.

FTP (File Transfer Protocol): Abbreviation of File Transfer Protocol, the protocol used on the Internet for transferring files.

Firewall: A system designed to prevent unauthorized access to or from a private network. Firewalls can be implemented in both hardware and software, or a combination of both. Firewalls are frequently used to prevent unauthorized Internet users from accessing private networks connected to the Internet, especially intranets. All messages entering or leaving the intranet pass through the firewall, which examines each message and blocks those that do not meet the specified security criteria.

Gateway: A gateway is a network point that acts as an entrance to another network.

HTTP: Acronym for Hypertext Transport Protocol the system for requesting HTML documents from the World-Wide-Web.

HTTPS: It's the security enhanced version of HTTP, and is used in the SSL of Internet servers. The example of HTTPS URL is: <https://www.abc.com>.

IMAP: Short for Internet Message Access Protocol, a protocol for retrieving e-mail messages. The latest version, IMAP4, is similar to POP3 but supports some additional features. For example, with IMAP4, you can search through your e-mail messages for keywords while the messages are still on the mail server. You can then choose which messages to download to your machine. Like POP, IMAP uses SMTP for communication between the e-mail client and server.

IPSec: Short for IP Security, a set of protocols being developed by the IETF to support secure exchange of packets at the IP layer.

IP address: Internet host address contains 32 bit digits.

ISP: Short for Internet Service Provider, a company that provides access to the Internet, some also providing services such as: disk renting, homepage making, mail service and server setup.

JVM: Acronym for Java Virtual Machine. An abstract computing machine, or virtual machine, JVM is a platform-independent programming language that converts Java bytecode into machine language and executes it. Most programming languages compile source code directly into machine code that is designed to run on a specific microprocessor architecture or operating system, such as Windows or UNIX. A JVM -- a machine within a machine -- mimics a real Java processor, enabling Java bytecode to be executed as actions or operating system calls on any processor regardless of the operating system.

Netstat: UNIX command (also used in Windows) to show the current TCP/IP connections and addresses.

NIC: Abbreviation of Network Information Center, provides services such as help, document and training etc for users. NIC has locations all over the world.

POP3: POP3 (Post Office Protocol 3) is the most recent version of a standard protocol for receiving e-mail. POP3 is a client/server protocol in which e-mail is received and held for you by your Internet server. Periodically, you (or your client e-mail receiver) will check your mail-box on the server and download any mail.

Proxy server: A server that sits between a client application, such as a Web browser, and a real server. It intercepts all requests to the real server to see if it can fulfill the requests itself. If not, it forwards the request to the real server.

Router: A device that connects any number of LANs. Routers use headers and a forwarding table to determine where packets go, and they use ICMP to communicate with each other and configure the best route between any two hosts.

SMTP: Short for Simple Mail Transfer Protocol, a protocol for sending e-mail messages between servers. Most e-mail systems that send mail over the Internet use SMTP to send messages from one server to another; the messages can then be retrieved with an e-mail client using either POP or IMAP.

Socket: Socket is a Circuit-level gateway. It applies security mechanisms when a TCP or UDP connection is established. Once the connection has been made, packets can flow between the hosts without further checking.

SSL: SSL (Secure Sockets Layer) is a commonly-used protocol for managing the security of a message transmission on the Internet. SSL uses a program layer located between the Internet's Hypertext Transfer Protocol (HTTP) and Transport Control Protocol (TCP) layers.

PKCS: The Public-Key Cryptography Standards (PKCS) are a set of intervendor standard protocols for making possible secure information exchange on the Internet using a public key infrastructure (PKI). The standards include RSA encryption, password-based encryption, extended certificate syntax, and cryptographic message syntax for S/MIME, RSA's proposed standard for secure e-mail.

Public Key: A public key is a value provided by some designated authority as a key that, combined with a private key derived from the public key, can be used to effectively

encryption messages and digital signatures. And public key can be distributed to the public.

Private Key: In cryptography, a private or secret key is an encryption/decryption key known only to the party or parties that exchange secret messages. A public key is used together with a private key. Private key is used to decrypt a message which has been encrypted with the public key, and to sign a message.

PKI: Short for public key infrastructure, a system of digital certificates, Certificate Authorities, and other registration authorities that verify and authenticate the validity of each party which is involved in an Internet transaction. It uses an asymmetric encryption algorithm for which the encryption key is different from the decryption key.

RC2: Short for Rivest Ciphers 2, is a symmetric encryption algorithm by Ron Rivest.

RSA: RSA is an Internet encryption and authentication system that uses an algorithm developed in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman. The RSA algorithm is based on the fact that there is no efficient way to factor very large numbers. It's an asymmetric encryption.

Shell: Command interpreter which accepts the input from user and forwards it the system. Shell generally refers to UNIX shell.

Protected Network: Protected networks are the TCP/IP networks on your intranet. Usually they are Local Area Networks (LANs) inside your firewall. The SRAC server guards those networks from unauthorized access from the Internet. Only SRAC users and visitors can access resources such as the SRAC Clients on the protected network. The SRAC server keeps a list of protected networks. An item on the list has two values, a subnet netid and a NIC IP address that has a routing path to the subnet. The NIC must be on the SRAC installed server hardware. The SRAC administrator must enter the list during the SRAC initialization.

Telnet: A terminal emulation program for TCP/IP networks such as the Internet. The Telnet program runs on your computer and connects your PC to a server on the network. You can then enter commands through the Telnet program and they will be executed as if you were entering them directly on the server console.

TCP/IP: Abbreviation of Transmission Control Protocol/ Internet protocol, an Internet protocol.

URL: A URL (Uniform Resource Locator) is the address of a file (resource) accessible on the Internet. The URL contains the name of the protocol required to access the resource, a domain name that identifies a specific computer on the Internet, and a hierarchical description of a file location on the computer.

VPN: A virtual private network (VPN) is a private data network that makes use of the public telecommunication infrastructure, maintaining privacy through the use of a tunneling protocol and security procedures. A virtual private network makes it possible to have the same secure sharing of public resources for data.

Reference 2: Standards Compliant in EverLink SRAC Server

	NAME	STANDARD	IMPLEMNET	
1	DES	FIPS PUB 46-3	ALL	Data Encryption Standard
2	3DES		ALL	Triple DES EDE
3	RC2	RFC 2268	ALL	RC2 Encryption
4	AES	AES PROPOSAL	ALL	AES Proposal: The Rijndael Block Cipher
5	RSA	RFC 2437	ALL	RSA Cryptography Specifications Version 2.0
6	SSL	INTERNET DRAFT	VERSION 3.0	Secure Sockets Layer version 3.0
7	PKCS1		ALL	RSA Encryption Standard
8	PKCS3		ALL	Diffie-Hellman Key-Agreement Standard
9	PKCS5		ALL	Password-Based Cryptography Standard
10	PKCS7		ALL	Cryptographic Message Syntax Standard
11	PKCS8		ALL	Private-Key Information Syntax Standard
12	PKCS10		ALL	Certification Request Syntax Standard
13	PKCS11		ALL	Cryptographic Token Interface Standard
14	PKCS12		ALL	Personal Information Exchange Syntax

15	PKI		PHASE3	Public Key Infrastructure
16	X.509	ITU-T X.509	ALL	Information Technology - Open System Interconnection – The Directory: Authentication Framework
17	X.500	ITU-T X.501	ALL	Information Technology – Open System Interconnection – The Directory: Models
18	ASN.1	ITU-T X.680	ALL	Information Technology – Abstract Syntax Notation One (ASN.1): Specification of Basic Notation
19	CMS	RFC 2630	ALL	Cryptographic Message Syntax
20	HTTP	RFC 2616	VERSION1.1	Hypertext Transfer Protocol – HTTP/1.1
21	POP3	RFC 1939	ALL	Post Office Protocol – Version 3
22	SMTP	RFC 821	ALL	Simple Mail Transfer Protocol
23	ARPA NTM	RFC 822	ALL	Standard for The Format of ARPA Internet Text Messages
24	MIME	RFC 1521	ALL	MIME (Multipurpose Internet Mail Extension) Part One: Mechanism for Specifying and Describing the Format of Internet Message Bodies