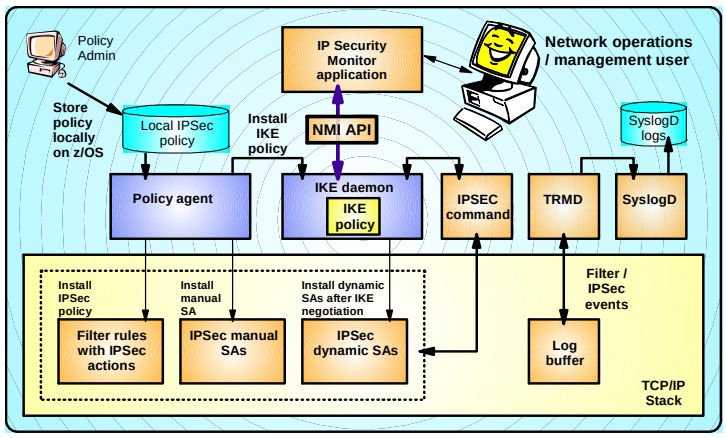

This presentation discusses the security enhancements in z/OS V1R9 Communications Server.

# IP Security network management interface (NMI)

- In z/OS V1R7, IP security network management data was made available with a UNIX® shell command (the ipsec command)
- Adding a formalized network management programming interface to retrieve IP security management data, enables a network management application, such as IBM Tivoli OMEGAMON® to access IP Security management data and integrate such data into the network management functions OMEGAMON already provide




Data similar to what can be retrieved using the ipsec command will be available over the IP Security NMI interface:

- IP filtering rules and statistics
- IKE Phase I SA information and status
- IKE Phase 2 SA information and status
- Manual SA information and status
- Port translation data

IBM Tivoli's OMEGAMON for Mainframe Network product is expected to utilize this interface.




In z/OS V1R7 Communications Server, IP security network management data was made available with a UNIX shell command (the ipsec command). Adding a formalized network management programming interface to retrieve IP security management data, enables a network management application to access IP Security management data and integrate such data into the network management functions already provided.

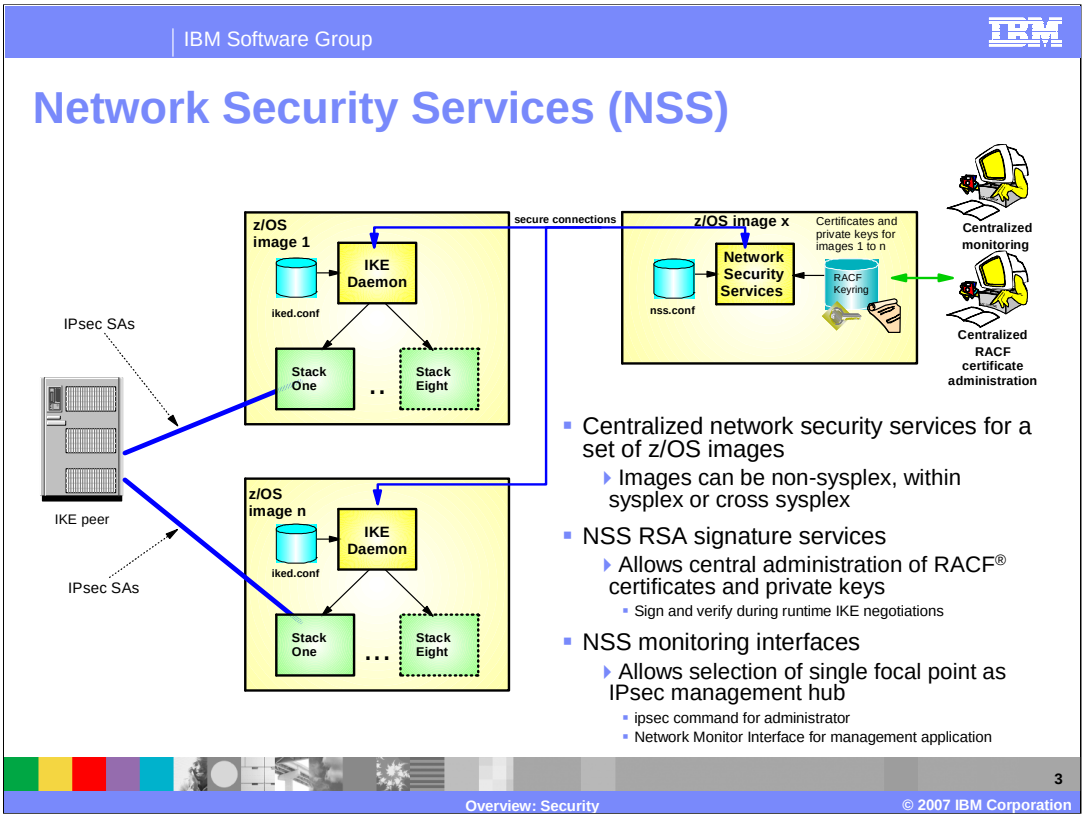
The ipsec command displays and manages system information for Integrated IPSec, but a programmatic interface was needed for network management applications to perform these actions without needing to resort to screen-scraping.

A network management interface is implemented for Integrated IPSec in z/OS V1R9 Communications Server. The IKE daemon implements an AF_UNIX listening socket that accepts connections, and uses a request/response model for providing IPSec management data and control. Consequently, IKED must be running in order to make use of this service. Data similar to what can be retrieved using the ipsec command will be available over the IP Security NMI interface:

- •IP filtering rules and statistics
- •IKE Phase I Security Association information and status
- •IKE Phase 2 Security Association information and status
- •Manual Security Association information and status
- •Port translation data

In addition the interface allows for the following:

- •Activation or deactivation of manual tunnels
- •Activation, deactivation, or refresh of IP tunnels
- •Deactivation or refresh of IKE tunnels
- •Load default IP filters or policy IP filters




Network Security Services (NSS) centralizes the sensitive keying material that would otherwise need to reside in less secure zones of the network onto a single location in the most secure zone of the network. In addition, NSS allows for centralized configuration and administration of certificates.

Network Security Services provide centralized certificate services, monitoring and management for IPSec security across z/OS systems within and across sysplexes. Network Security Services allow IPSec certificates to be kept in a single location, rather than having them reside on each z/OS node. The z/OS Communications Server IKE daemon is enhanced so that it can be configured to act as a Network Security client. Configuration is on a per-stack basis, such that each NSS-enabled stack will appear to the Network Security Server as an independent client. For TCP/IP stacks that are not configured to use Network Security Services, the IKE daemon will continue to manage certificates out of a local keyring.

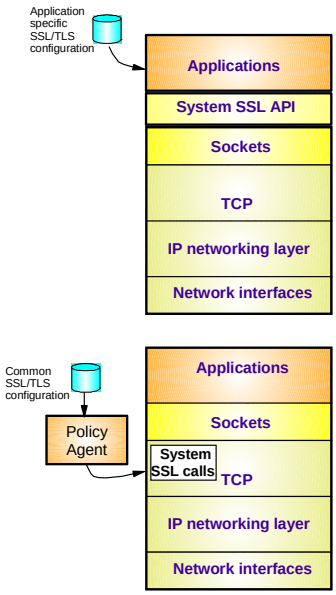
Specifically, NSS provides a central SAF-enabled repository for RSA certificates along with signature services within the most trusted zones. It eliminates the need to distribute certificates to security endpoints. NSS centralizes and reduce configuration and deployment complexity, especially when used along with Centralized Policy Services. It offloads digital signature operations from IKE daemon (the NSS client) and it enables monitoring and management of remote IPSec endpoints through the ipsec command and a network management programming interface.

The network security services (NSS) server provides a set of network security services for IPSec. These include the certificate (and digital signature) service and the network management service. The certificate service and network management service are used by NSS clients. When an NSS client uses the NSS certificate service, the NSS server creates and verifies RSA signatures on the behalf of the NSS client using RSA certificates that are stored only at the NSS server. When an NSS client uses the network management service, the NSS server routes IPSec network management interface (NMI) requests to that NSS client, which enables the NSS client to be managed remotely. The NSS client provides the NSS server with responses to these requests.

As mention earlier, the IKE daemon can be configured to act as an NSS client on behalf of multiple TCP/IP stacks. A separate connection is maintained to the server for each NSS-enabled TCP/IP stack, so each TCP/IP stack appears as a separate NSS client to the NSS server. The -z option of the ipsec command or the IPSec NMI can be used to manage NSS clients that use the NSS network management service. For details about using the ipsec command to manage NSS clients, see *z/OS Communications Server: IP System Administrator's Commands*. For details about using the IPSec NMI to manage NSS clients, see *z/OS Communications Server: IP Programmer's Guide and Reference*.

# AT-TLS API enhancements

- SSL/TLS support can since z/OS V1R7 be implemented using one of two methods on z/OS:
  - Change applications that need SSL/TLS security to support SSL/TLS configuration and invoke system SSL directly.
  - Let the Application-Transparent TLS (AT-TLS) layer inside the TCP/IP stack handle all SSL/TLS processing
- Using AT-TLS has several advantages:
  - AT-TLS provides SSL/TLS features above and beyond what most SSL/TLS applications choose to support
  - Support of new SSL/TLS functions, such as new ciphersuites, can be added without application changes
- Enhancements to the AT-TLS API
  - Stop security on a connection
  - Allow both secure and non-secure connections to the same port







SSL/TLS support can, since z/OS V1R7. be implemented using one of two methods on z/OS:

•Change applications that need SSL/TLS security to support appropriate application-specific configuration options to specify SSL/TLS options, such as keyring, cipher suites, application-specific security options, and so on. Replace selected socket calls with calls to system SSL (C/C++/Java only)

•Let the Application-Transparent TLS layer inside the TCP/IP stack handle all SSL/TLS processing. This method provides common SSL/TLS configuration for all applications through an AT-TLS policy (managed by the Policy agent). It uses optimized SSL/TLS code within the TCP/IP stack that interfaces to system SSL to implement the SSL/TLS functions. In many cases SSL/TLS support can be added without application changes

Using AT-TLS has several advantages:

•AT-TLS provides SSL/TLS features above and beyond what most SSL/TLS applications choose to support - such as, support for Certificate Revocation Lists (CRLs), multiple keyrings per server, optional use of system SSL cache, and so on.

•AT-TLS uses an optimized SSL/TLS infrastructure that performs better than when SSL/TLS is implemented directly in the applications.

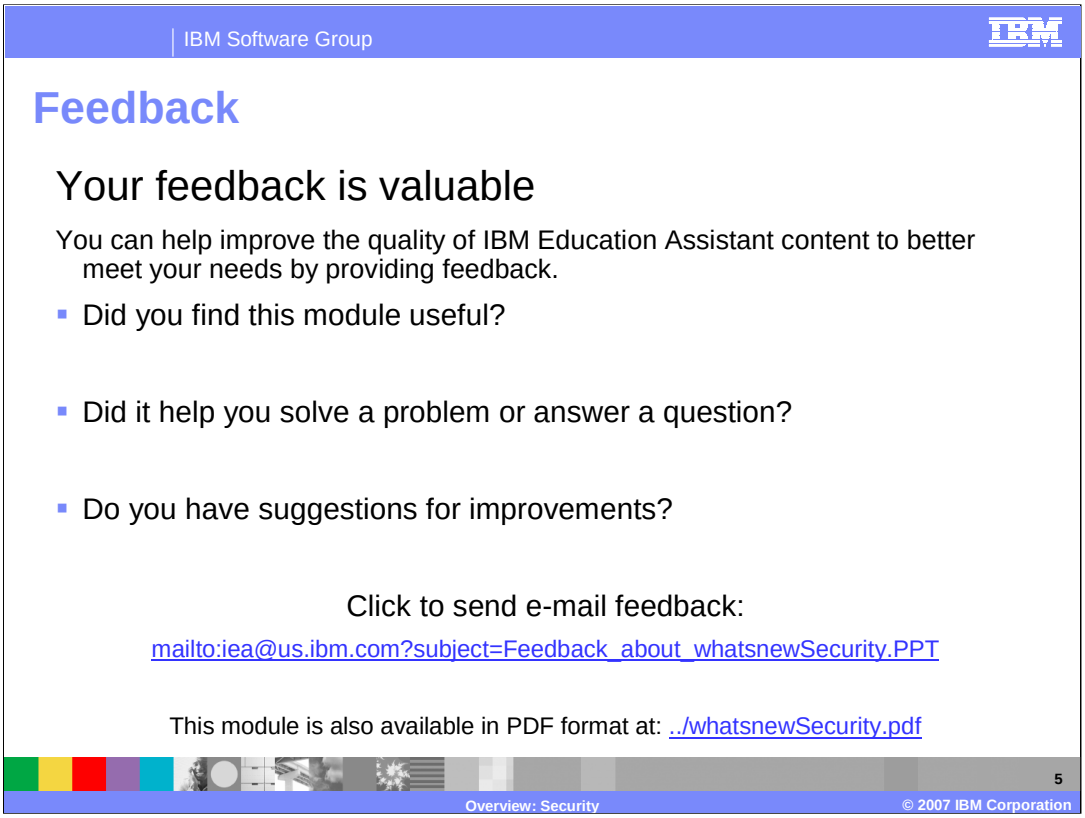
•Support of new SSL/TLS functions, such as new ciphersuites, can be added without application changes.

•Allows SSL/TLS-enabling non-C sockets applications on z/OS, such as CICS Sockets, Assembler- and Callable sockets, and so on.

z/OS 1.7 introduced Application Transparent Transport Layer Security (AT-TLS) and the SIOCTTLSCTL IOCTL. This allowed applications to control AT-TLS security on a connection. The application starts security on the connection. The application can also reset the cipher being used to generate new session keys for the connection or reset the session associated with the connection to force a full SSL handshake. This type of application is called a controlling application. The AT-TLS policy must be defined with ApplicationControlled On.

Many applications use a secure connection for sensitive data during the connection. After this data exchange, security is no longer needed for the connection. The application will stop security on the connection, reducing the CPU overhead of security. Some applications also support both secure and non-secure connections on the same port. These applications detect which type of client has connected and act accordingly. These type of applications could not use the SIOCTTLSCTL ioctl to implement security.

Two new options now are defined for the SIOCTTLSCTL ioctl. TTLS_Stop_Connection allows the application to stop security on a connection. The SSL security on the connection will be stopped and future data will be sent as clear text. TTLS_Allow_HSTimeout will allow the SSL handshake to timeout if no SSL data is received from the client or if clear text data is received. This option is only valid with TTLS_Init_Connection since it only applies to a SSL handshake on a clear text connection.




You can help improve the quality of IBM Education Assistant content by providing feedback.

# Trademarks, copyrights, and disclaimers

The following terms are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM OMEGAMON RACF z/OS

UNIX is a trademark of The Open Group in the United States, other countries, or both.

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

Information is provided "AS IS" without warranty of any kind. THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.