TCP/IP Services for OpenVMS: Security

Mark T. Hollinger

TCP/IP Services Engineering
Hewlett-Packard Company

HP WORLD 2003
Solutions and Technology Conference & Expo

Agenda

- Security terminology
- Internet security technologies available:
  - Kerberos
  - SSL (Secure Sockets Layer), can protect POP
  - SSH (Secure Shell) setup walk-through
  - IPsec (future)
  - DNSSEC

Security and cryptography terminology

- Security
  - Authentication
    - to confirm the identity of users (or hosts)
  - Integrity
    - to guarantee that data are unaltered
  - Privacy
    - encryption to provide confidentiality
- Cryptography
  - Secret key
    - symmetric
    - single, shared key
  - Public key
    - asymmetric
    - public and private keypairs
    - digital signatures

Threat models

What are you protecting against?

- Insiders, on isolated network or behind firewall
  - If you trust everyone on your network, security is easy
  - Just watch for worms and viruses
- Active attacks
  - Often require sophisticated techniques
  - Intruder runs risk of detection
- Passive attacks
  - Opportunistic eavesdropping
  - Can only capture data which happens to appear
  - Password snatching (plaintext passwords are bad)

Kerberos V2 on OpenVMS 7.3-2

- Scheme for mutual authentication and optional data encryption, developed at MIT
- A port of KerberosV5 MIT release 1.2.6
- Features:
  - 64 bit and 32 bit interfaces
  - KRB4 to KRB5 support
  - Improved documentation (GSSAPI's documented)
  - Triple DES
- Packaging change
  - Separate PCSI installable kit
  - Provides flexibility to address CERTs and fixes
- Still supports "Kerberized" TCP/IP Services TELNET

Configuring Kerberos

- Follow Kerberos installation instructions
  - Kerberos is covered in OpenVMS documentation set
  - Join or establish a realm
  - If needed, generate and distribute host keys
  - Ensure principals exist for users and applications

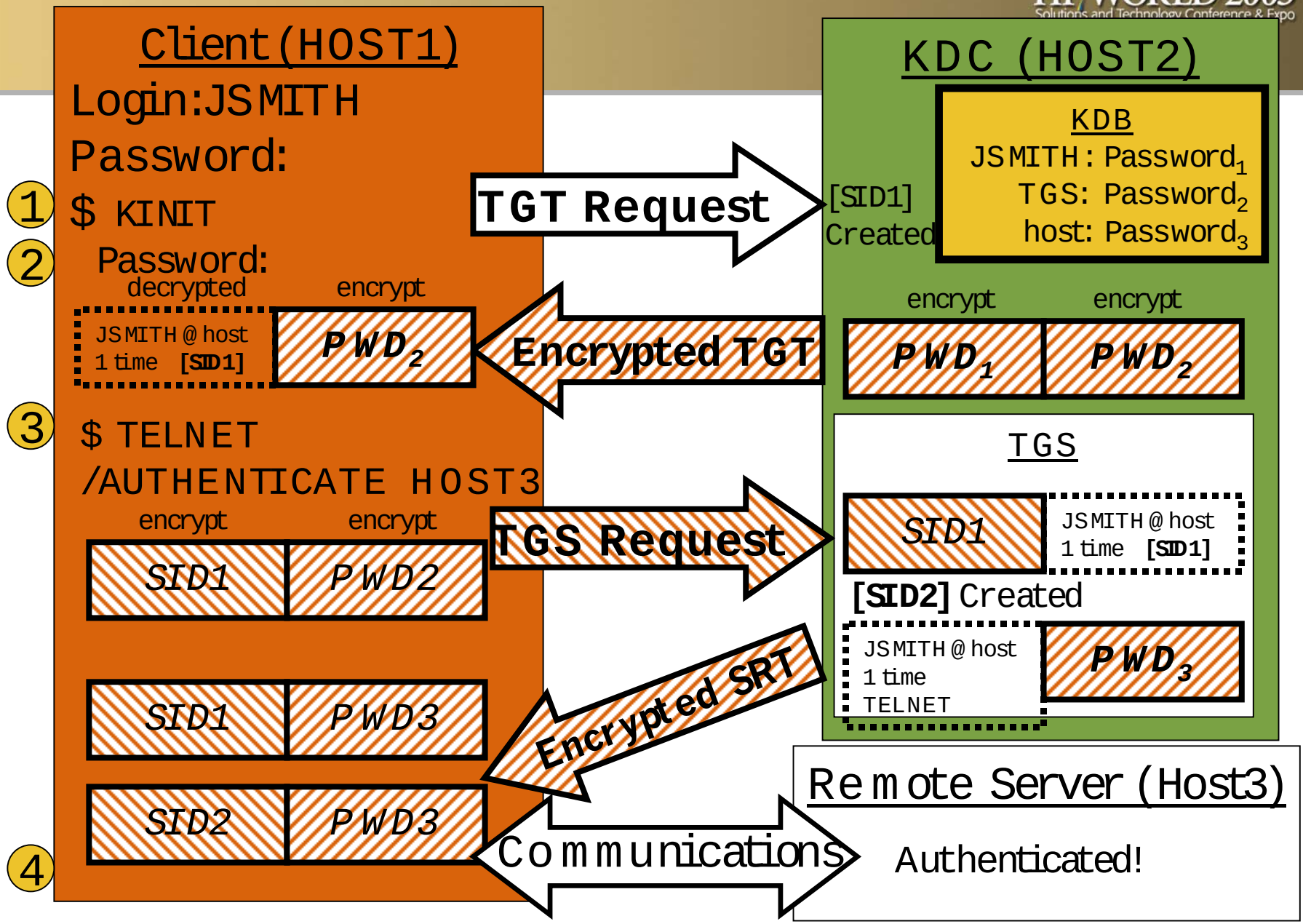
- Execute SYS$MANAGER:TCPIP$CONFIG.COM
  - Select “Optional components” menu
  - Add Kerberos for TELNET server

Using Kerberos

- Obtain credentials
  - Host1_$ KINIT
  - Password for jsmith@REALM.EXAMPLE.COM:

- Login using credentials instead of password
  - TELNET /AUTHENTICATE HOST2 2323
  - %TELNET-I-TRYING, Trying … 16.1.2.3
  - %TELNET-I-ESCAPE, Escape character is ^]
  - [Kerberos V5 accepts "jsmith@REALM.EXAMPLE.COM"]
  - Host2_$

Kerberos ticket process



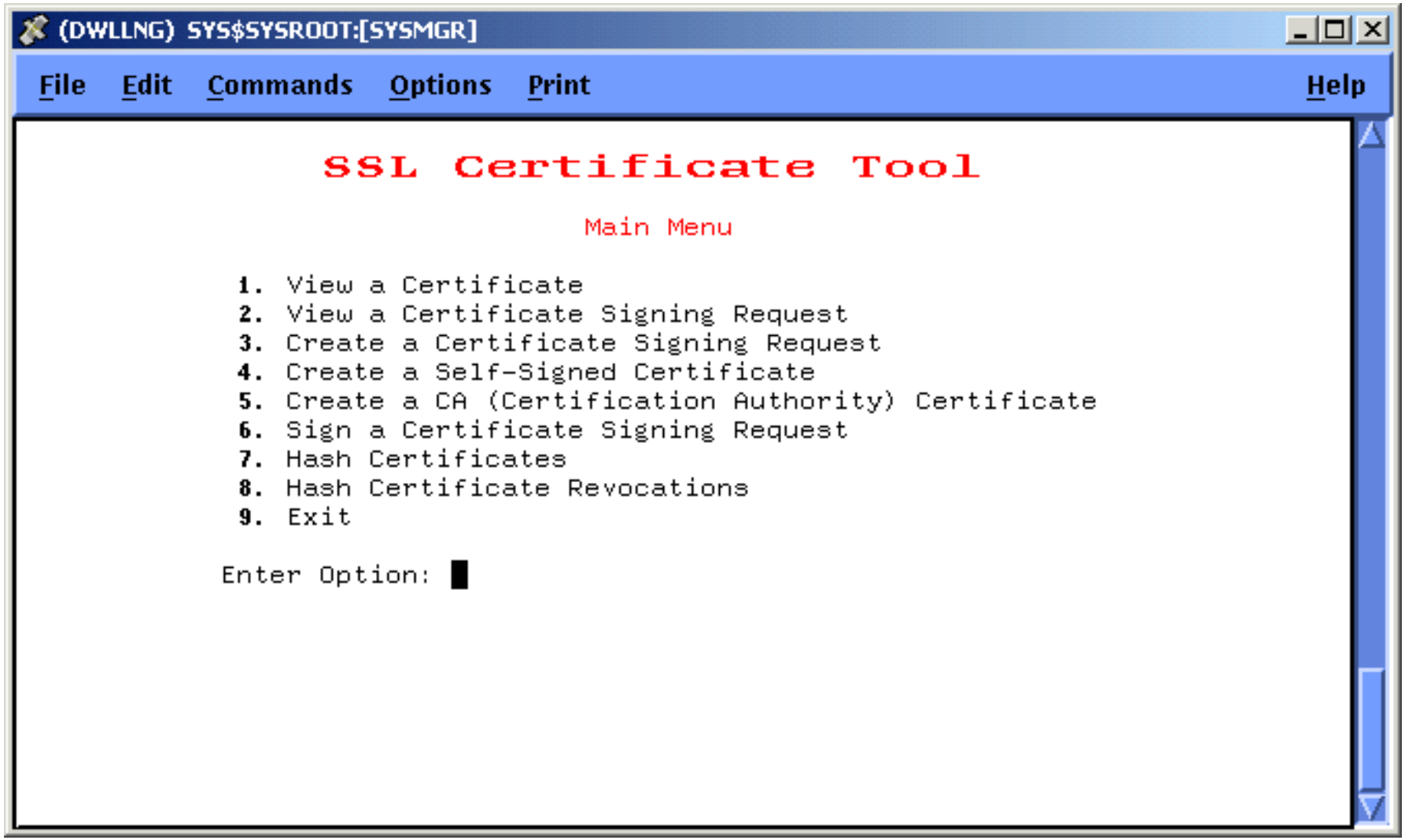
What is SSL?

- Used by secure (https://…) web servers
  - widely deployed
- Certificate based
- SSL protects privacy and offers server authentication
- Server setup required
  - easy, using self-signed certificates
  - use CA-issued certificates to boost security
- Requires no special client configuration
  - does not (usually) provide mutual authentication
- Can secure other protocols (e.g., POP, TELNET)

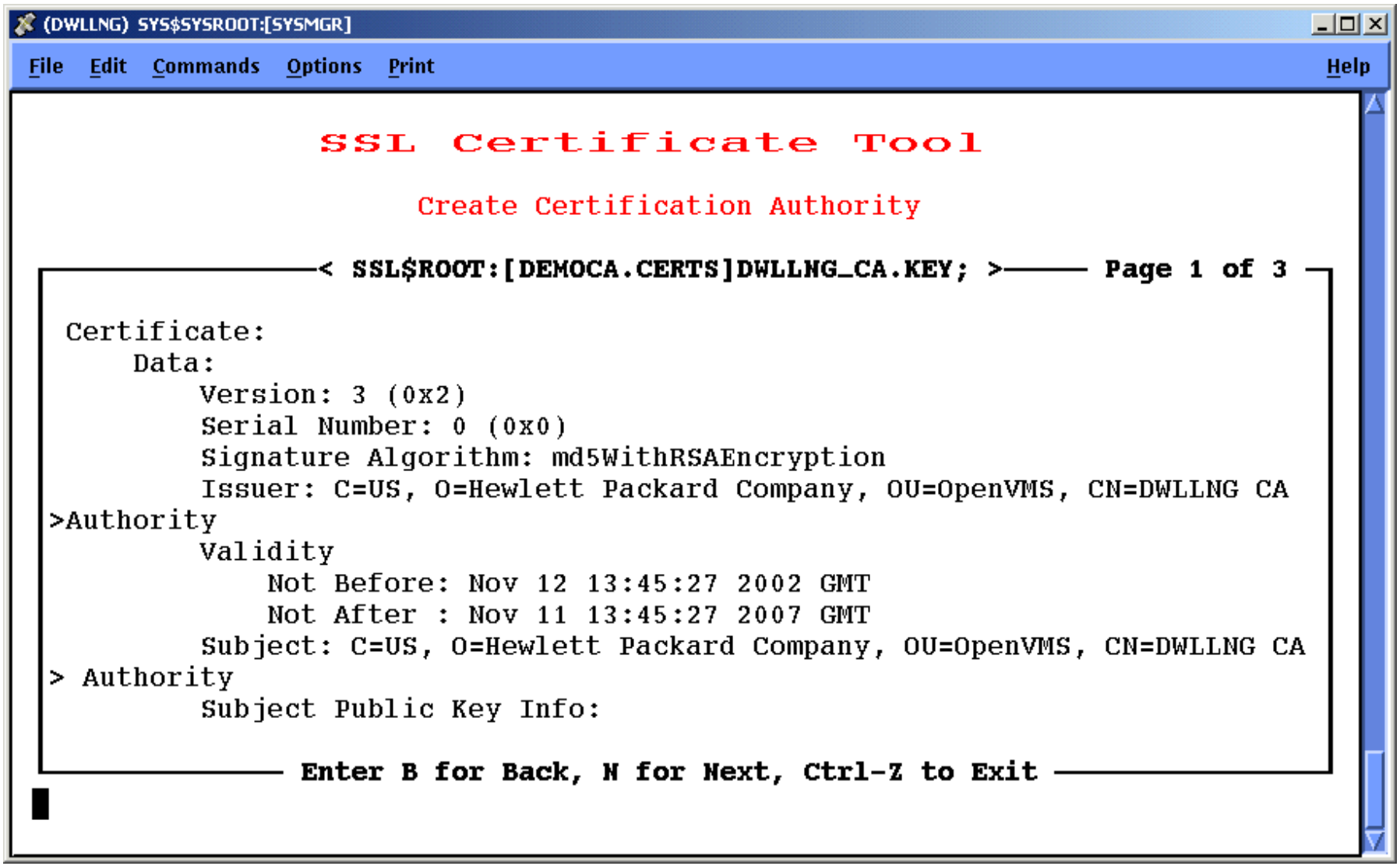
SSL for OpenVMS V1.1

- Port of OpenSSL V0.9.6G with all Security Patches
- Includes OpenSSL features:
  - Threading package
  - Independent utilities
  - 64-bit API calls
  - Certificate tool
  - Improved documentation!
  - *Open Source Security for OpenVMS Alpha Vol 2: Compaq SSL (Secure Sockets Layer) for OpenVMS Alpha*

Certificate tool - $ @SSL$COM:SSL$CERT_TOOL



Display Certificate Authority certificate



Secure POP server with SSL

- New Feature in TCP/IP Services V5.4
- POP server accepts SSL connections on port 995
- Passwords and mail are no longer sent in the clear
- Many clients, including Outlook [Express], support this
- Requires OpenVMS SSL kit
- No need for a separate, unsupported Stunnel process

- Similar IMAP enhancement planned for future release
  - Note: IMAP SSL remains available via Stunnel

What is SSH?

- Application-level security solution
- SSH secures:
  - Terminal sessions, file copy, and remote command execution
  - Other protocols via port forwarding:
    POP, FTP, X, SMTP, IMAP, even VPNs
- Consists of client, server, and support programs
- SSH is de-facto standard, turned Internet standard
- Available on many platforms

SSH components

SSH consists of:

• SSH	Login client
• SSHD	SSH2 server
• SSH-KEYGEN	Key generation facility
• SSH-AGENT	Holds keys in memory
• SSH-ADD	Maintains keys inside agent
• SSH-SIGNER	Digital key signer
• SCP/SFTP	File transfer applications

SSH capabilities

Features in the first release:

- Remote logins (yes)
- File transfer (stream_lf)
- Remote command execution (yes)
- Key generation and agents (yes)
- Port forwarding (yes)
- Authentication (password, host, key) (yes)
- Multiple encryption algorithms (yes)
- X forwarding (manual)

SSH (secure shell)

- Ported from Tru64 UNIX® Version 5.1B
- Developed by SSH Communication Security, Inc.
- Secure Shell (SSHv2) – V5.3 EAK available now
  - Supports SSHv2 secure connections
  - Supports SCP secure file transfer
- Fully integrated in TCP/IP Services for OpenVMS V5.4:
  - Configurable using TCPIP$CONFIG
  - Managed through UNIX-style commands
  - Compatible with OpenVMS auditing and access control
  - Uses ASCII configuration files (same as UNIX)

SSH authentication setup

- Password authentication is easy
  - Just enable SSH via TCPIP$CONFIG
- Public-key authentication requires manual steps:

```
$ @SYS$MANAGER:TCPIP$DEFINE_COMMANDS
$ SSH_KEYGEN
Generating 1024-bit dsa key pair
   1 oOo.oOo.oOo
Key generated.
Passphrase :
Again      :
Private key saved to ssh2/id_dsa_1024_a
Public key saved to ssh2/id_dsa_1024_a.pub
$ CREATE [.SSH2]AUTHORIZATION.
key id_dsa_1024_a.pub
*Exit*
$ CREATE [.SSH2]IDENTIFICATION.
idkey id_dsa_1024_a
*Exit*
```

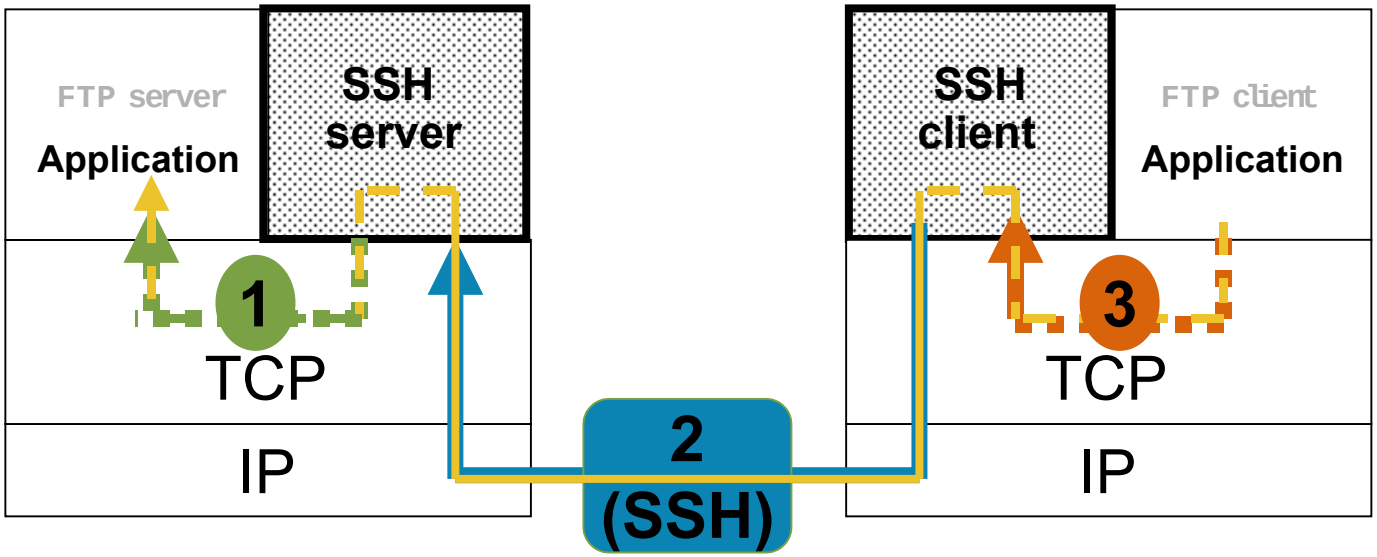
SSH authentication (cont.)

- Next, copy files in [.SSH2] from client to server:
  - ID_DSA_1024_A.PUB
  - AUTHORIZATION.
  - Avoid plaintext FTP if possible
- Once files are in target user's [.SSH2] on server:

```
$ SSH SERVER -L JJONES
Passphrase for key "ssh2/id_dsa_1024_a" with comment "1024-bit dsa,
jsmith@server.ucx.lkg.dec.com, Tue Jul 22 200303:43:55":
Authentication successful.
$
```

- Didn't work? Try checking:
  - SYS$SYSDEVICE:[TCPIP$SSH.SSH2]SSH2_CONFIG.
    (should allow publickey authentication method)

SSH port forwarding (FTP example)

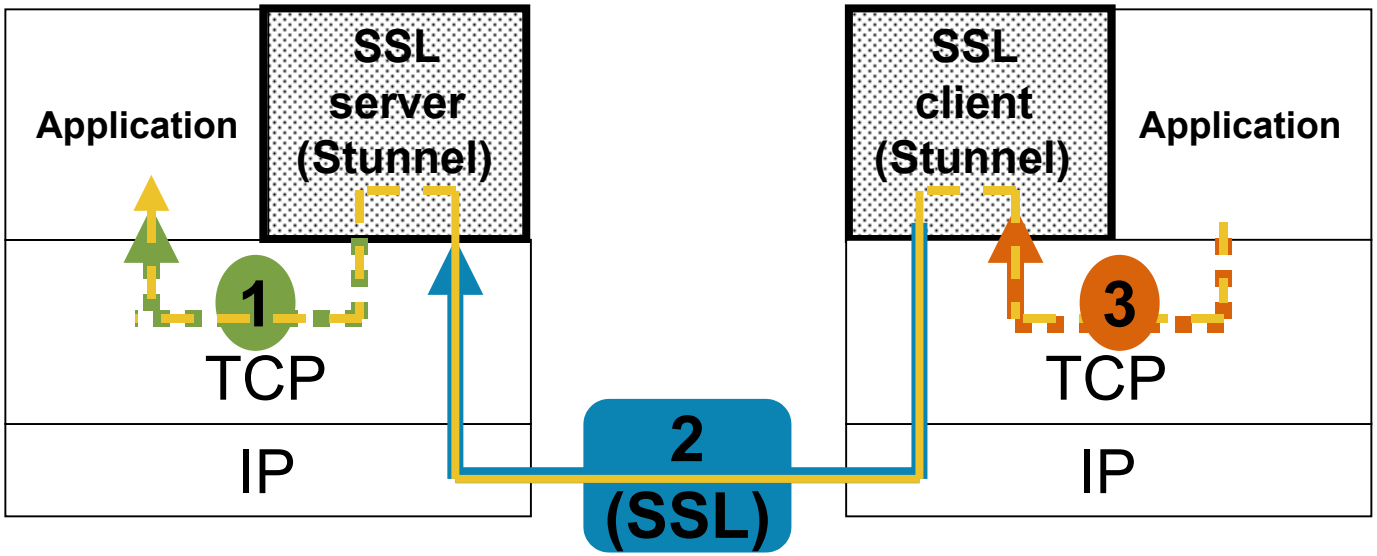


1. SSH server: (no action required; already running)
2. SSH client: (ssh remote "-Lftp/2222:localhost:21")
3. Application: (ftp localhost 2222)

Stunnel (secure tunnel)

- Open source freeware
- Allows you to encrypt arbitrary TCP connections inside an SSL connection
- Communicate with any remote SSL application
- Secures non-SSL aware applications
  - TELNET, FTP, RCP, IMAP, etc.
  - Stunnel provides the encryption
  - Requires no changes to the original application

Stunnel (TELNET example)



1. SSL server: (stunnel -d 992 -r localhost:23 -p stunnel.pem)

2. SSL client: (stunnel -c -d 992 -r remote:992)

3. Application: (telnet localhost 992)

What is IPsec?

- Standards based IP-level solution to security
- IPsec secures everything above IP
- Provides:
  - ESP (Encapsulated Security Payload)
  - AH (Authentication Header)
  - IKE (Internet Key Exchange)
- Security policy dictates what is encrypted and what algorithms are available during IKE dialog
- When selected, can protect every packet
- Work for both for IPv4 and IPv6
- Support planned for future TCP/IP Services release

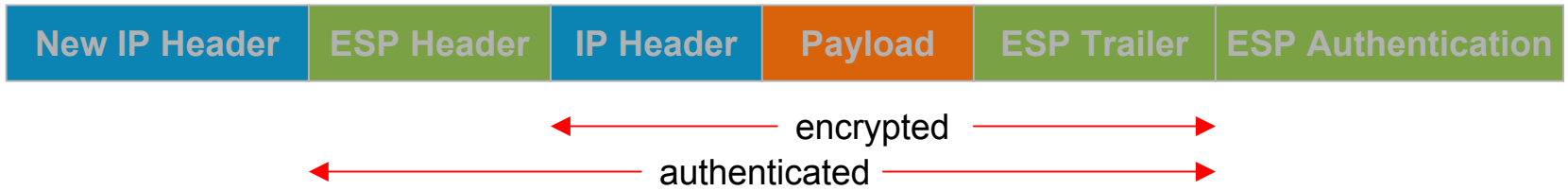
Authentication Header (AH)

- Provides data integrity and authentication of origin between two systems
  - No confidentiality
  - Designed to be algorithm-independent
    - Required authentication algorithms: Keyed MD5 or SHA-1
- Tunnel mode
  - Packet is appended to tunnel header and AH header

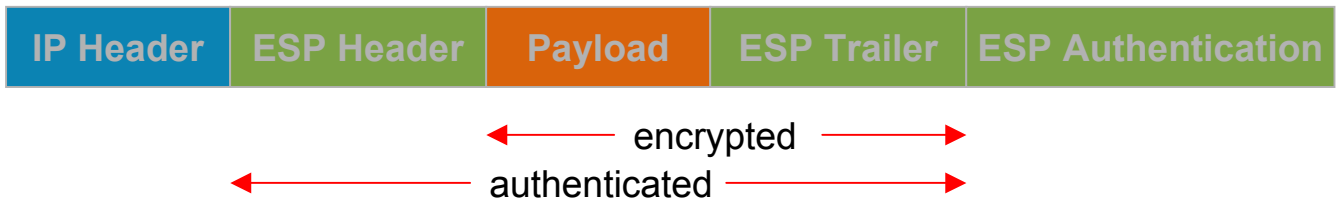
- Transport mode
  - Original packet's IP header is the IP header of resulting packet

Encapsulating Security Payload (ESP)

- Provides data integrity, authentication, and confidentiality
  - Significant performance impact
- Tunnel mode
  - Encrypted packet is appended to tunnel header



- Transport mode
  - Original packet's IP header is the IP header of resulting packet



IPsec components

- Complex set of protocols, mechanisms, and tools
  - Engine: processes incoming and outgoing packets, real time
  - Interceptor: interface to the engine
  - Policy Manager: maintains a security policy DB
- Applications:
  - Digital certificate utilities
  - Cryptographic utilities
  - LDAP utilities
- ISKAMP/IKE: Security Association and Key Management

IPsec vs. secure application layer

- SSH, SSL/TLS
  - Usually built into each application
  - Controlled by the application
  - Only applies end-to-end
- IPsec
  - Applies to all network traffic
  - Controlled by the system administrator
  - Part of network infrastructure (VPNs)

DNS security

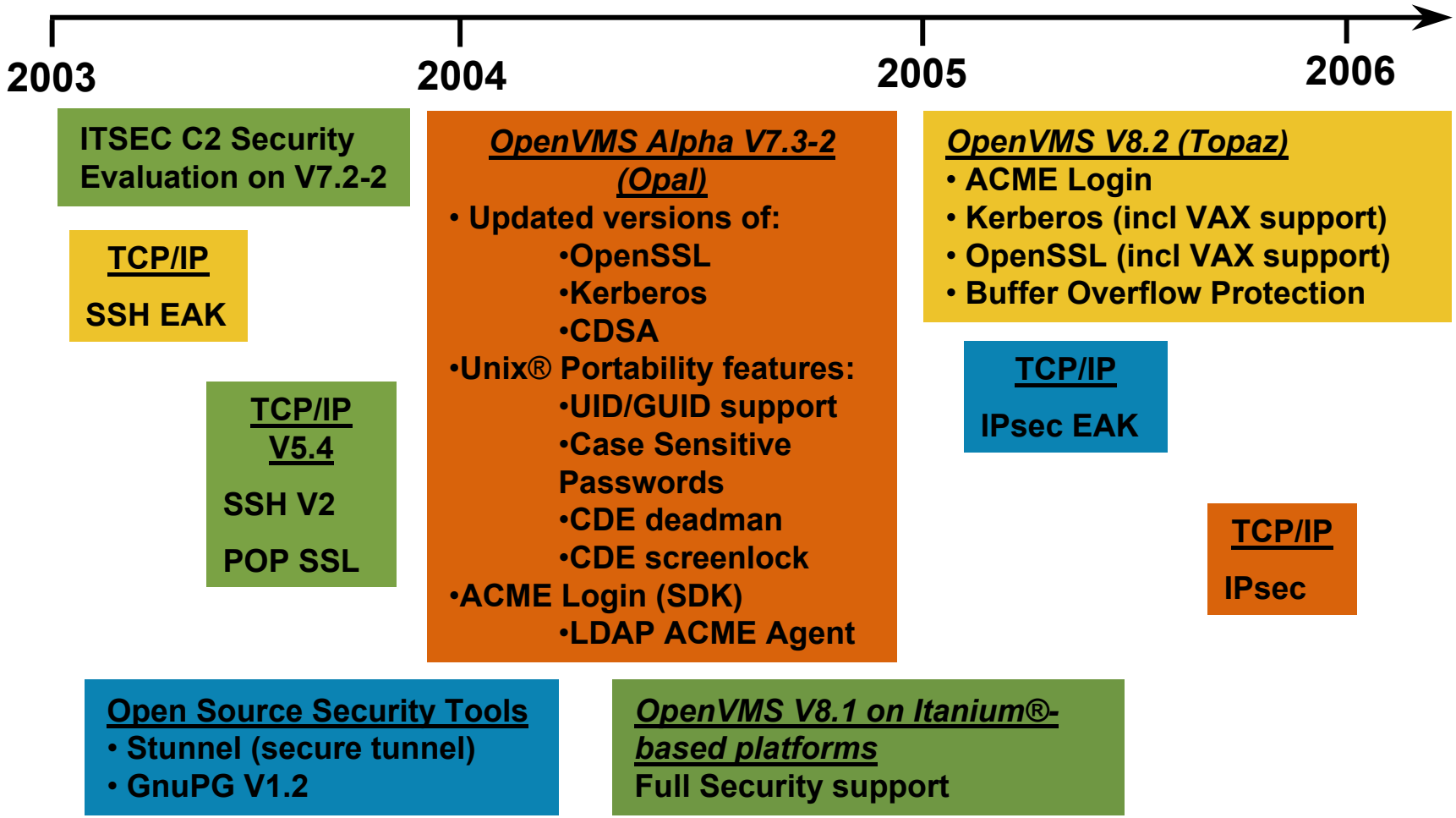
- TSIG (Transaction SIGnature)
  - Signs DNS messages, such as dynamic updates
  - Uses shared secret
- “Simple” to configure
  - Does not scale well
- Part of BIND 8.2
- DNSSEC is a set of protocols to authenticate the data returned by DNS name servers
- Somewhat complicated to set up DNSSEC
- Partial DNSSEC support: part of BIND v9 code

GnuPG 1.2 port to OpenVMS

- GnuPG is a complete and free replacement for PGP. GnuPG is Free Software. It can be freely used, modified and distributed under the terms of the GNU General Public License.
- PGP, on which OpenPGP is based, was originally developed by Philip Zimmermann; see his page for background information on PGP.

- GnuPG (GNU Privacy Guard) is GNU's tool for secure communication and data storage. It can be used to encrypt data and to create digital signatures. GnuPG includes an advanced key management facility and is compliant with the proposed OpenPGP Internet standard.

OpenVMS Security Roadmap



Interex, Encompass and HP bring you a powerful new HP World.

Supplemental Materials

The following slides are not part of the presentation.

They are available for reference purposes.

Using Kerberos

- $ kinit
- Password for BAE@WEST.BUNDER.COM:
- $ klist
- Ticket cache: krb$user:[tmp]krb5cc_2293763
- Default principal: BAE@WEST.BUNDER.COM
- Valid starting Expires Service principal
- 13 Feb 01 22:13:28 14 Feb 01 08:13:23 krbtgt/WEST.BUNDER.COM@WEST.BUNDER.COM
- $ kdestroy and $ klist
- klist: No credentials cache file found while setting cache flags (ticket csche krb$user:[tmp]krb5cc_2293763)

Microsoft vs. MIT Kerberos

- Microsoft Kerberos is similar but not identical to MIT Kerberos
  - Windows Domain Controller must be the KDC
  - Microsoft Kerberos tickets contain additional fields that MIT Kerberos does not understand, but will ignore
  - Microsoft: http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/prodtechnol/windows2000serv/deploy/walkthru/kerbstep.asp

Kerberos Startup & Shutdown

- System Management Tools
  - SYS$STARTUP:KRB$STARTUP.COM
  - SYS$STARTUP:KRB$SHUTDOWN.COM
- Definitions should be added to system files
  - Add @SYS$MANAGER:KRB$LOGICALS to SYS$MANAGER:SYSTARTUP_VMS.COM
  - Add @SYS$MANAGER:KRB$SYMBOLS to SYS$MANAGER:SYLOGIN.COM

Configuring Kerberos

- The initial step in setting up a Kerberos system is running the configuration utility
  - @SYS$STARTUP:KRB$CONFIGURE
  - This will
    - Create the Kerberos server database (server)
    - Set up the administrative entries (client & server)
    - Define logical names

Kerberos administration

- Kerberos DCL support is available through:

  $ KERBEROS [/ADMIN]
  [/INTERFACE=[DECWINDOWS | CHARACTER_CELL]]

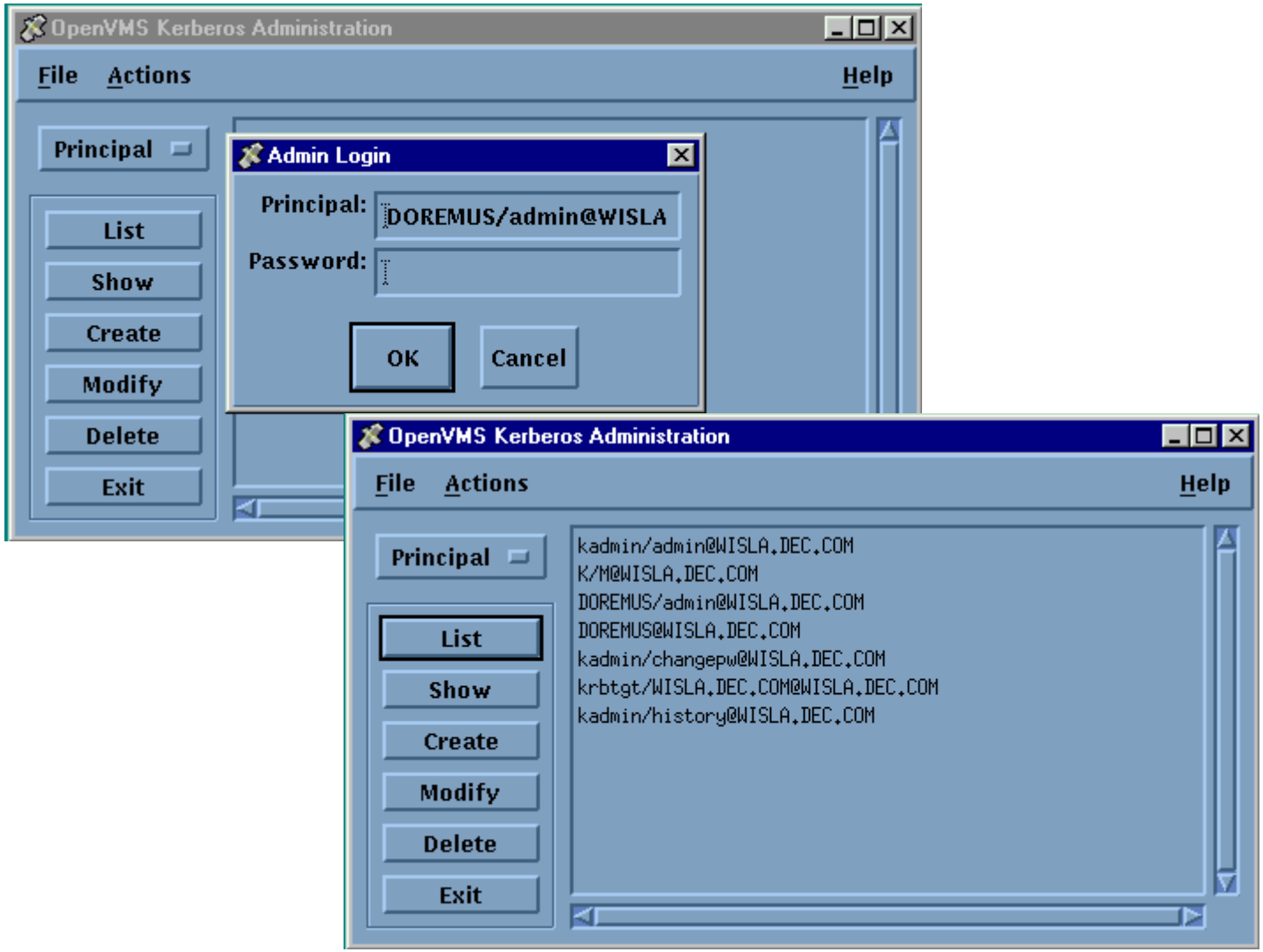
- **Adding Users**

- Before users can log in to Kerberos, principal entries must be created for each user

  $ KERBEROS /ADMIN
  KerberosAdmin> LOGIN
  Enter password:
  KerberosAdmin> CREATE PRINCIPAL newuser
  /PASSWORD=SecretPassword

OpenVMS GUI KDC

$ Kerberos /admin /interface=decwindows



SSL & OpenSSL

- Netscape developed SSL V2 & V3 protocols
- Transport Layer Security (TLS) is RFC 2246
- OpenSSL is a toolkit that provides:
  - SSLv2 & v3 protocols
  - TLS v1 protocol
  - Cryptographic algorithms
- OpenSSL is packaged as
  - an SSL library
  - a cryptographic library
  - a command line utility

Compaq SSL on OpenVMS Alpha V1.1

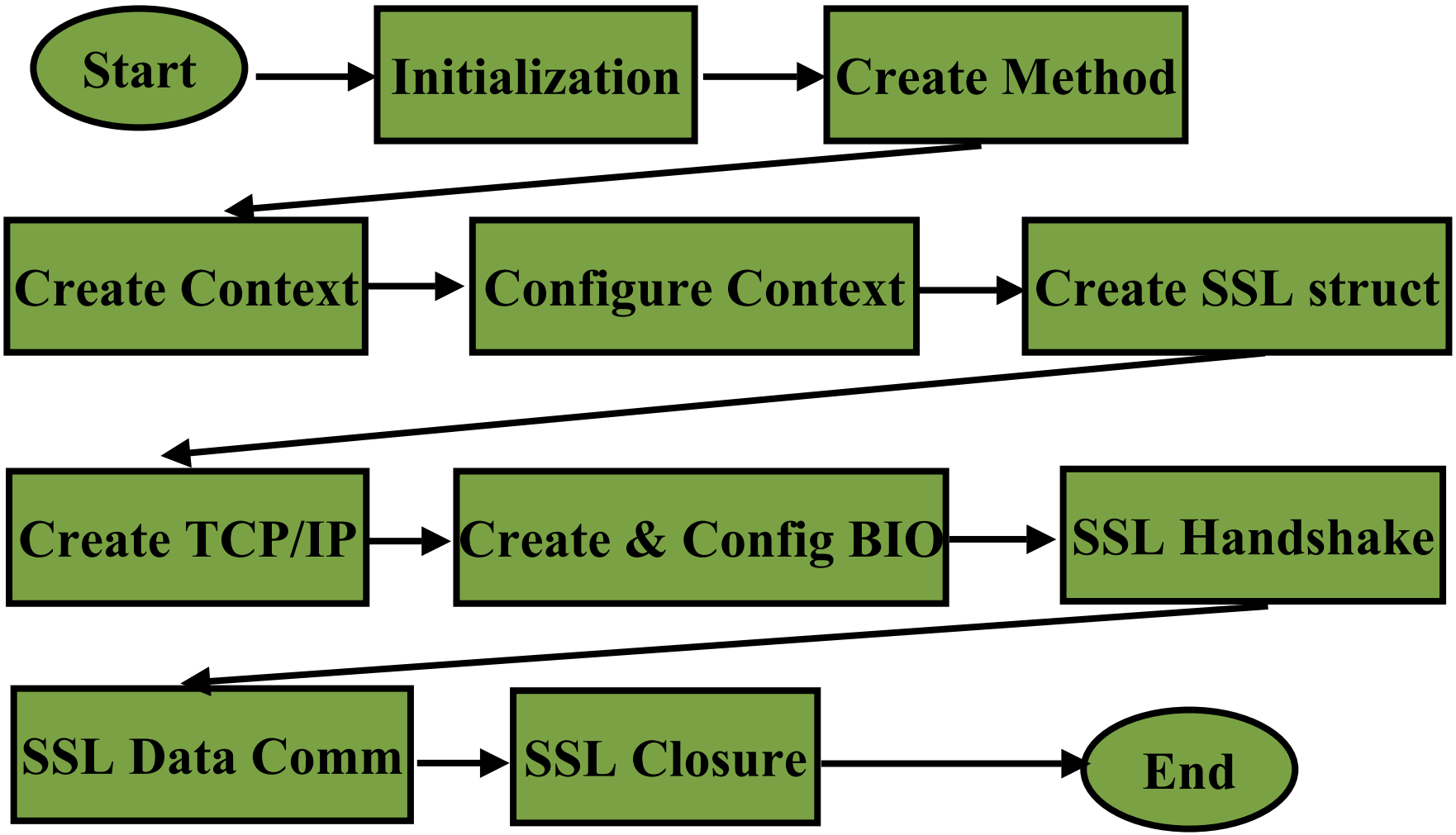
- From OpenSSL 0.9.6g baselevel and security updates
  - www.openvms.compaq.com/openvms/products/ssl/ssl.html
- OpenVMS enhancements sent back to OpenSSL Group
- Layered Product kit (.PCSI)
- Installation steps:

$ product install ssl[/dest=dev:[dir]]

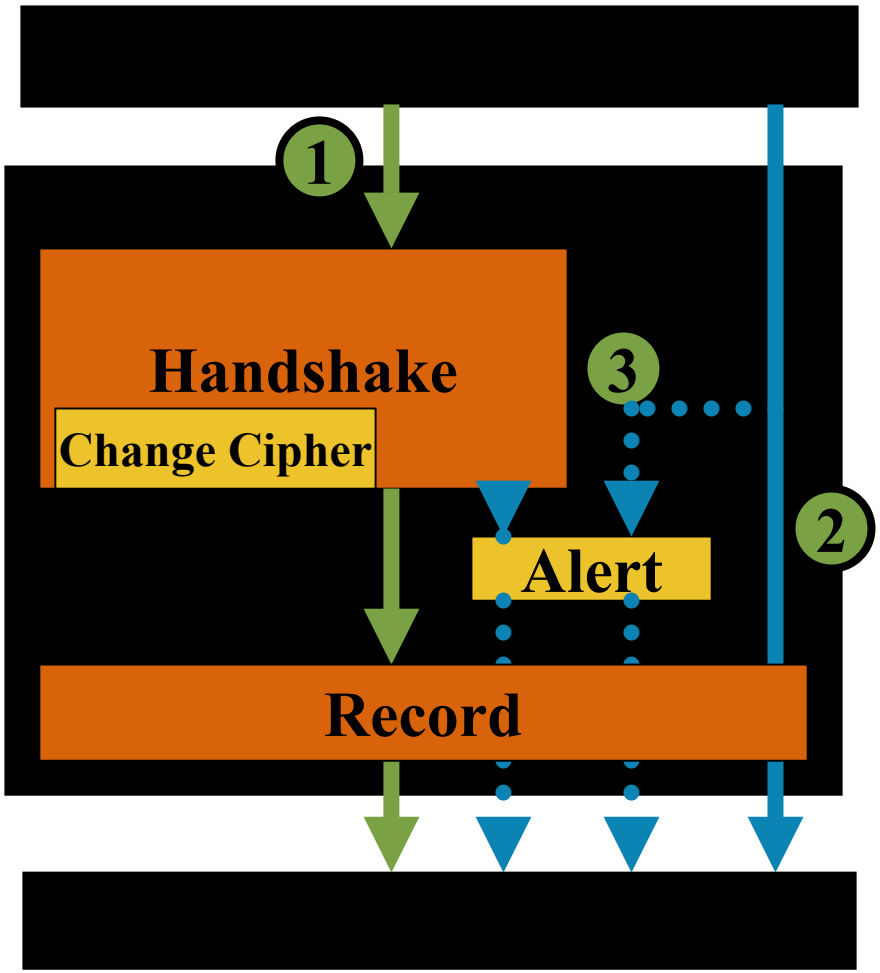
$ @sys$startup:ssl$startup

$ @ssl$com:ssl$utils

Overview of an SSL application

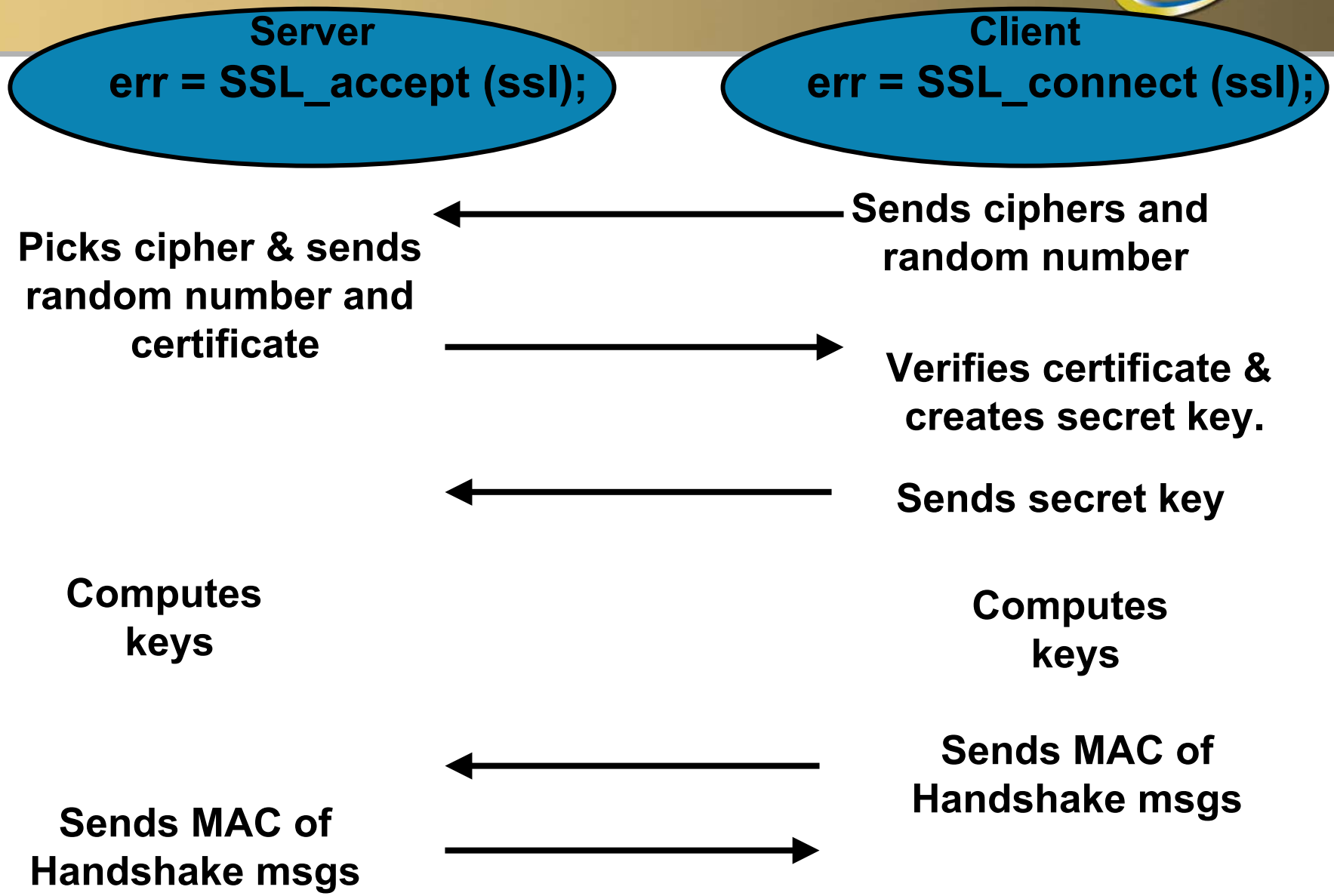


SSL/TLS protocol overview

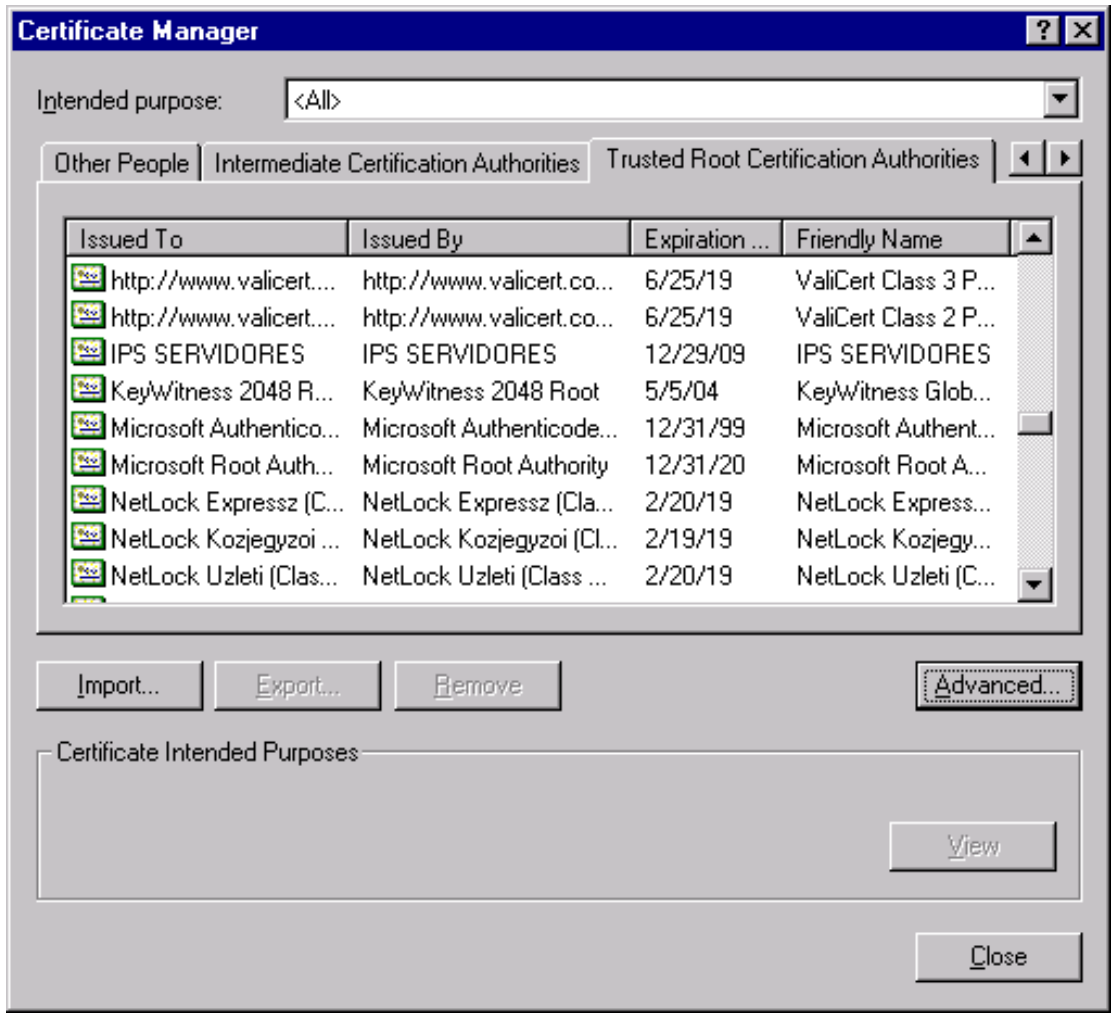


- **Handshake**
  - Establish shared secret for encryption
- **Record**
  - Encryption & data integrity for SSL
- **Alert**
  - Signaling errors & SSL closure
- **Change Cipher Spec**
  - Notify that cypro algorithms & keys are being changed

Handshake


Server
err = SSL_accept (ssl);
Client
err = SSL_connect (ssl);
Sends ciphers and random number
Picks cipher & sends random number and certificate
Verifies certificate & creates secret key.
Sends secret key
Computes keys
Computes keys
Sends MAC of Handshake msgs
Sends MAC of Handshake msgs

CA certificates in a browser (pre-loaded)




hp
invent