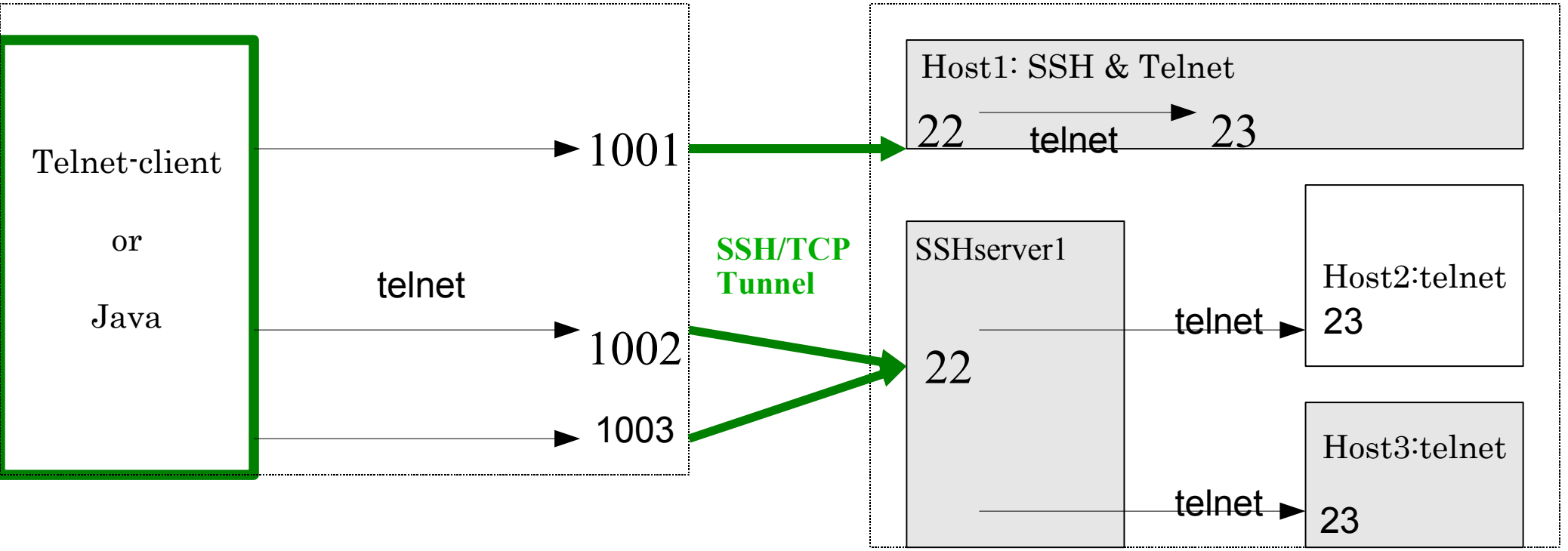
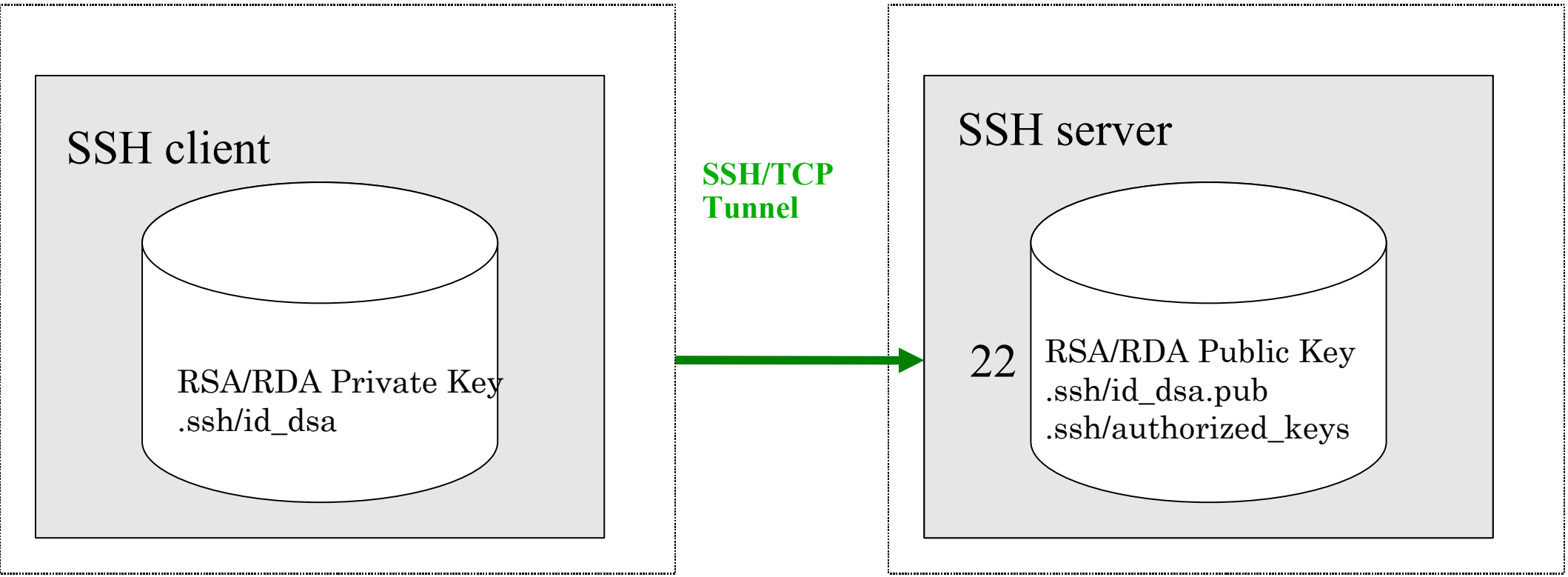
Telnet using SSH Local Port Forwarding



Use Java Apache org.apache.commons.net.telnet.telnetClient

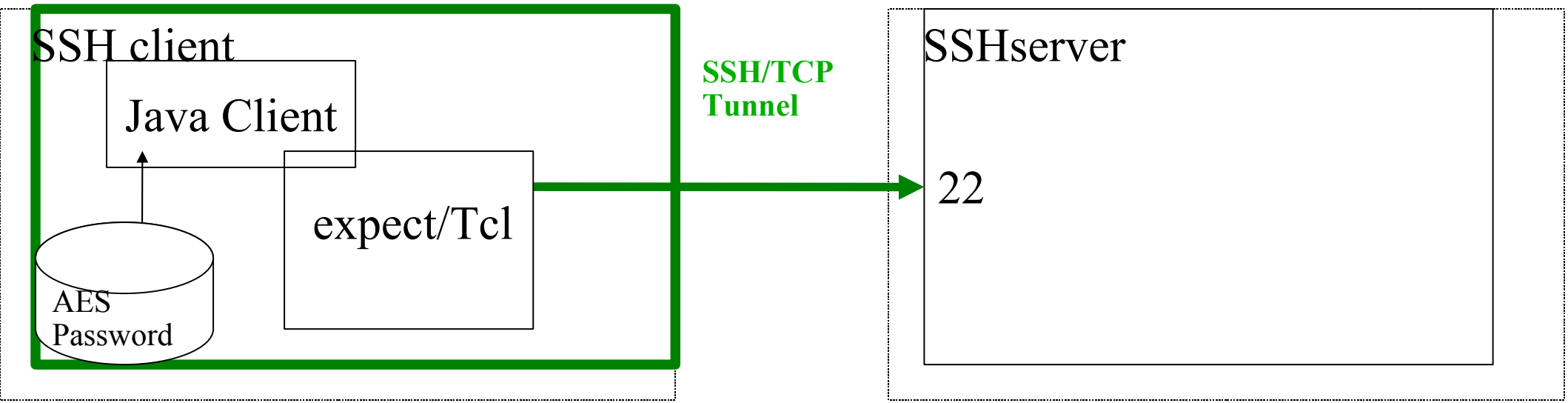
```
ssh -l userid  host1       -p22 -N -f -L1001:host1:23
telnet localhost 1001

ssh -l userid  SSHserver1 -p22 -N -f -L1002:host2:23
telnet localhost 1002
```

SSH Authentication Setup with RSA/DSA Keys



ssh-keygen -t dsa (Generate public/private key pair)
Move private_key to SSH client
Move public_key to SSH server

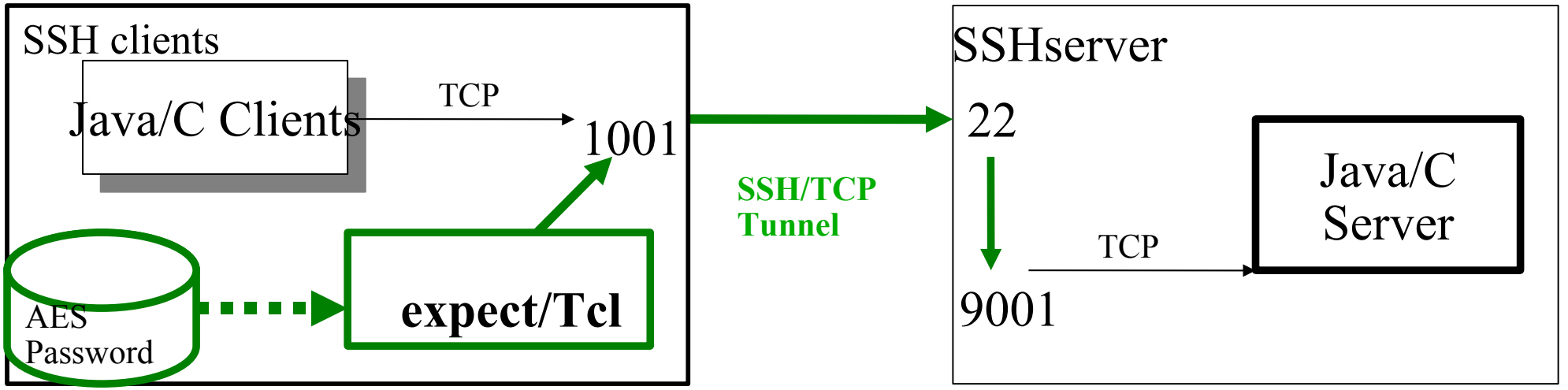
Java with automated foreground SSH session



Java Client:
- Decrypt user/password from an AES encrypted file
- Invoke expect/Tcl script using java.lang.ProcessBuilder()

expect/Tcl:
- Spawn: ssh -l userid server -p22
- Responds user/password for SSH server login
- Does keep-alive timeout to monitor SSH session
- Traps SSH/TCP disconnect/failure to notify JavaClient
- Passes through data between Java Client and SSH server

SSH Tunneling for Distributed Java/C Processes



expect/Tcl: sets up SSH/TCP Tunneling in background
- Spawn: ssh -l userid server -p22 -N -f -L1001:server:9001
- Decrypt user/password from an AES encrypted file
- Responds user/password for SSH server login
- Does keep-alive timeout to monitor SSH session
- Traps SSH/TCP disconnect/failure for SSH restarts

Java/C Clients: (existence of SSH is transparent)
- Do TCP connect to localhost/1001 to reach remote Java/C Server

Java/C Server: (existence of SSH is transparent)
- Listens TCP at 9001. Accept TCP connection requests.

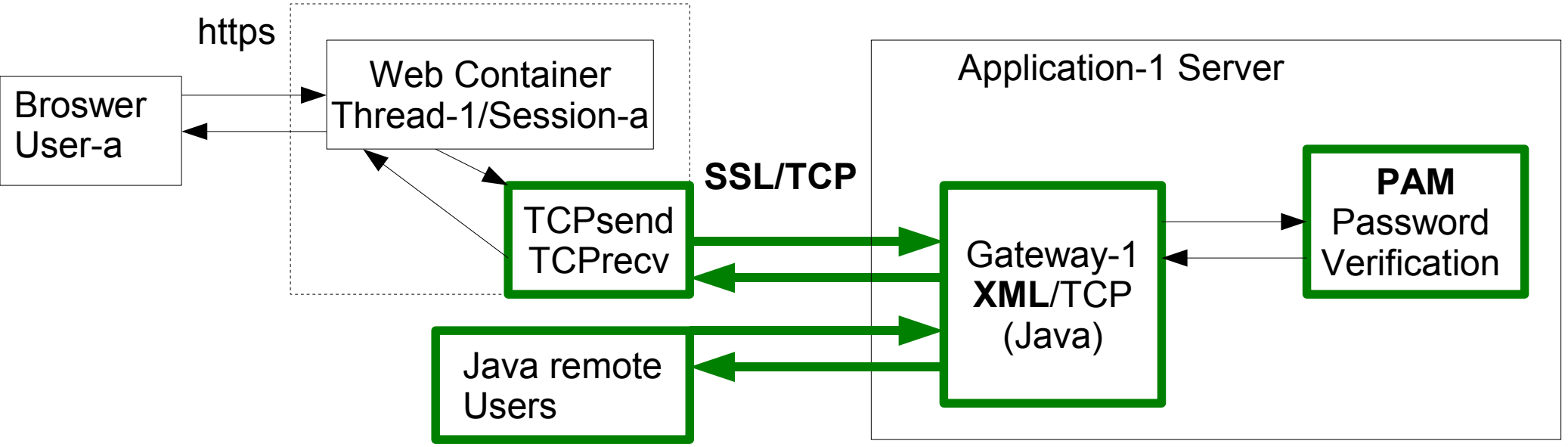
Java JDK/JCE AES Encryption

- **Install JDK/JRE "unlimited Strength Version JCE" for AES 128+ bit support**
- **Designed & coded Java classes to perform AES encryption & decryption as application infrastructure software**
- **Generate AES private/symmetric keys with keys size ranges from 128, 192, 256, etc.**

Java SSL/TCP Design & Coding

- **Designed & coded Java classes to perform SSL/TCP client and server functions as application infrastructure software.**
- **Generate RSA SSL non-symmetric key pair (keystore/certificate) for Tomcat-server & browser users.**
- **Generate RSA SSL non-symmetric key pair (keystore/certificate) for Java-server & Java-client applications.**
- **Designed & coded a Java TCP monitor which translates http/https to facilitate SSL TCP debugging & data trace**

Users Authentication Using Unix PAM



UNIX PAM: Pluggable Authentication Modules
Struts/JSP GUI sends user/password thru Thread-1 for TCPsend
- **Gateway-1 forward user/password to the PAM Verification process**
- **The C process verifies user/password with Appl Server thru PAM**
- **The Authentication result is sent back thru TCPrecv to Browser**

Each backend application may use its original Authentication for new Struts/JSP based GUI users.
Web Container Authorization/Authentication, if used, is independent and transparent to the application.