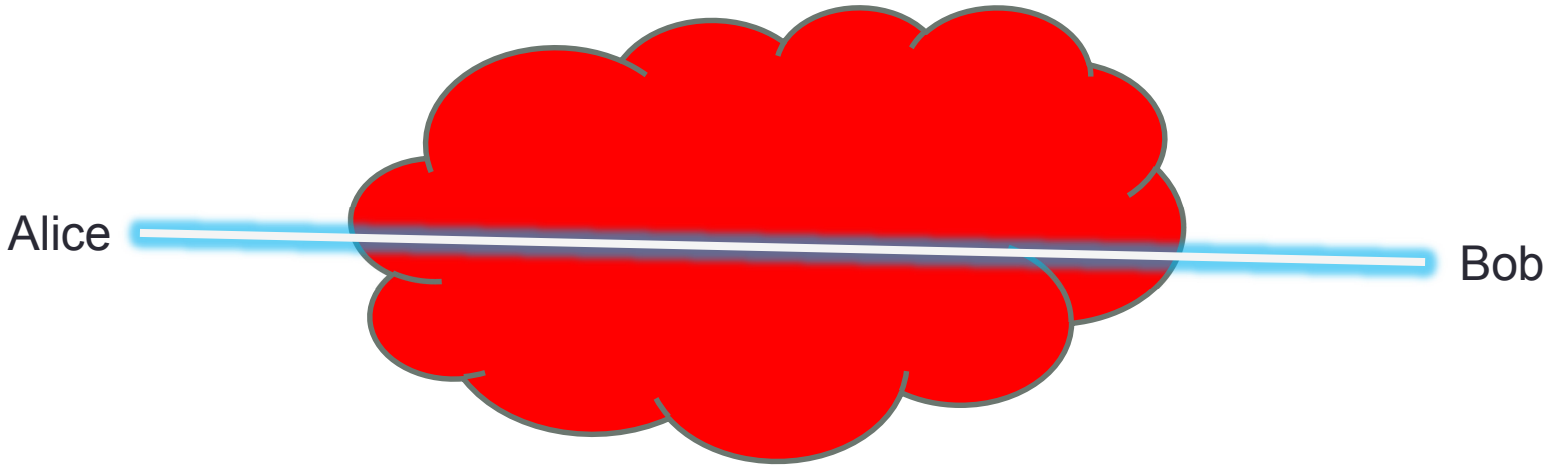
NETWORK SECURITY PRACTICES – ATTACK AND DEFENSE

End-to-End Encrypted Link

Network Security Approaches



Network Security Protocols

- SSL/TLS
  - Secure sockets layer / Transport layer security
  - Used mainly to secure Web traffic
- SSH
  - Secure Shell
  - Remote login
- IPsec
  - IP-level security suite

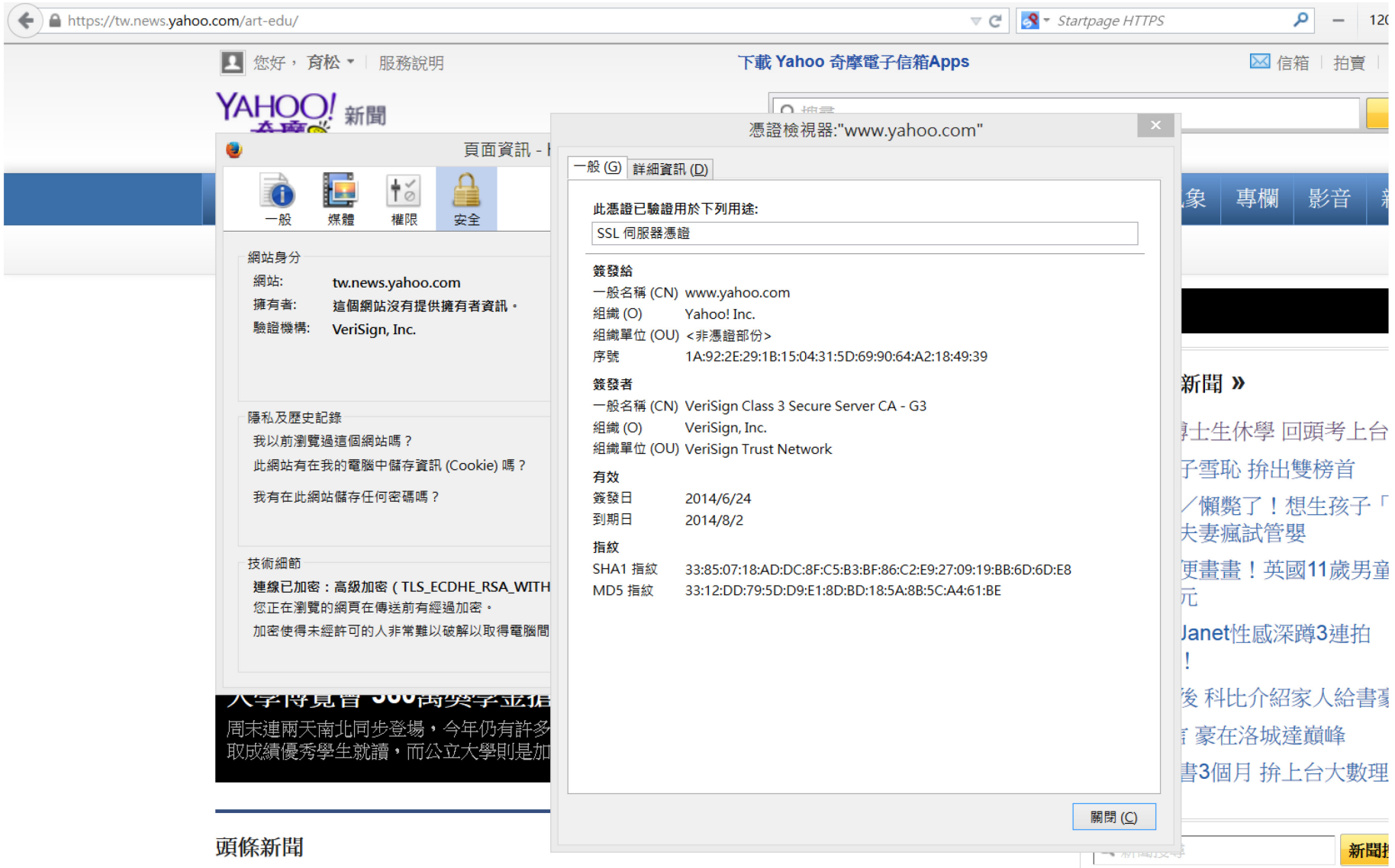
SSL

- Mid ‘90s introduced concerns over credit card transactions over the Internet
- SSL designed to respond to thse concerns, develop e-commerce
- Initially designed by Netscape, moved to IETF standard later

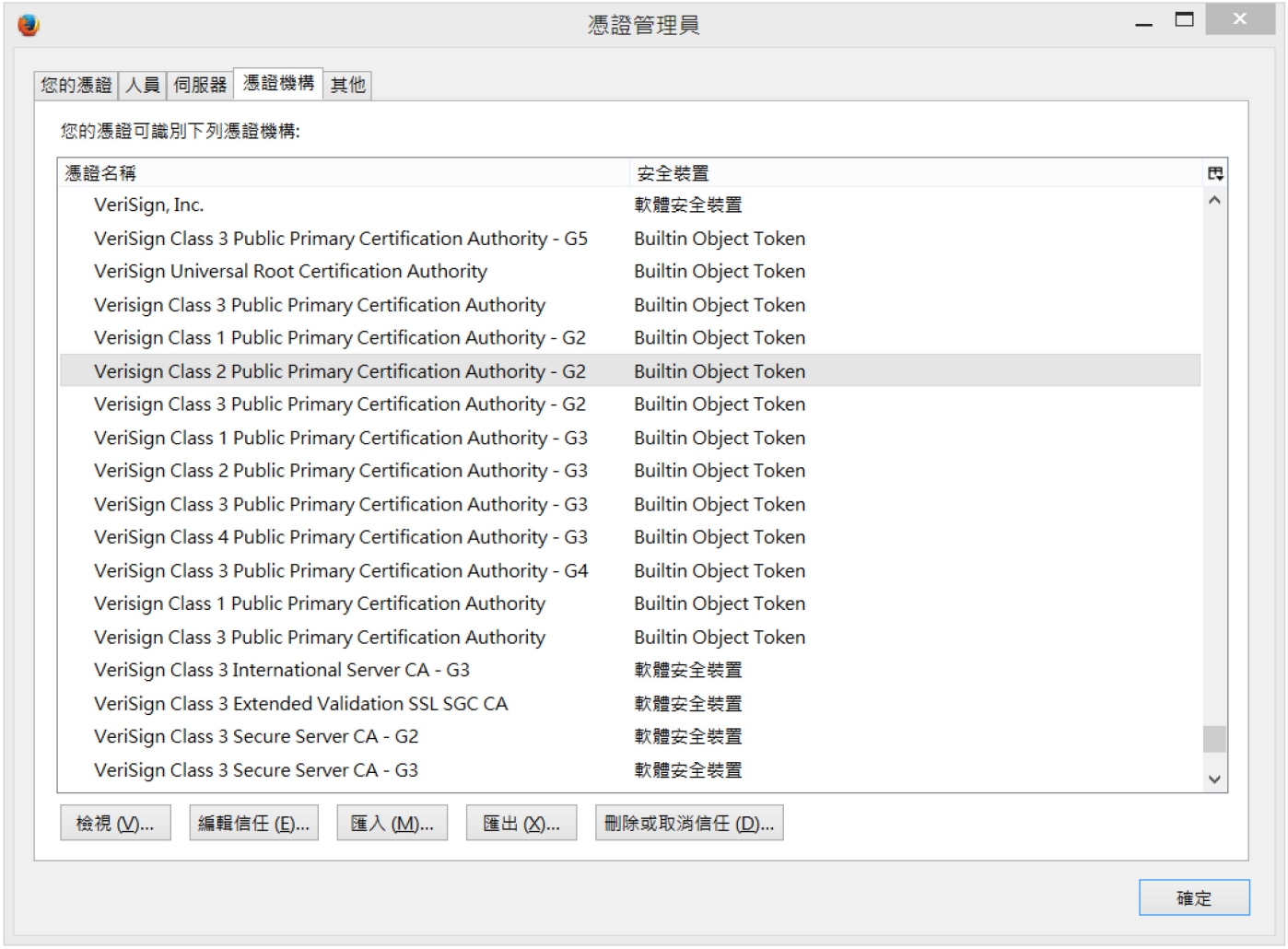
SSL model

- A client and a server
- Implements a socket interface
  - Any socket-based application can be made to run on top of SSL
- Protect against:
  - Eavesdroppers
  - MITM attacks
- Server has X.509 certificate
  - Client may have a certificate, too

SSL certificates (server certificate)



SSL certificates (CA certificates)



SSL certificates (client certificate)

SSL history

- SSLv2 1994
- SSLv3 1996
  - Fixed security problems
- TLS v1.0 1999
- TLS v1.1 2006

SSL key lengths

- Earlier versions used 40-bit keys for export reasons
- Later versions switched to 128-bit keys, with an option to use 40-bit ones with legacy servers/clients
- Rollback attack:
  - MITM

SSL sequence

- Negotiate parameters
- Key exchange
- Authentication
- Session

SSL negotiation

- Choice of cipher suites, key exchange algorithms, protocol versions
- E.g. : choice of 40- or 128-bit keys for export reasons
- Rollback attack: MITM chooses least secure parameters

SSL key exchange

- Diffie-Hellman key exchange
- RSA-based key exchange
  - Encrypt secret *s* with public key of server
  - (resists attacks described in last class)

Reaction attack

- Send encrypted nonce, see if it is of the right form
  - If padding decrypts incorrectly, server sends an error
  - Otherwise, SSL protocol continues
- Use algebraic properties of RSA to deduce bits of key from error
- Fixed with better padding
- Better fix: don’t send error
  - Pretend key exchange was OK, continue protocol with bogus session key

SSL authentication

- Anonymous (no authentication)
- RSA authentication (implicit)
- Sign Diffie-Hellman parameters
  - Achieves *perfect forward secrecy*

SSL session

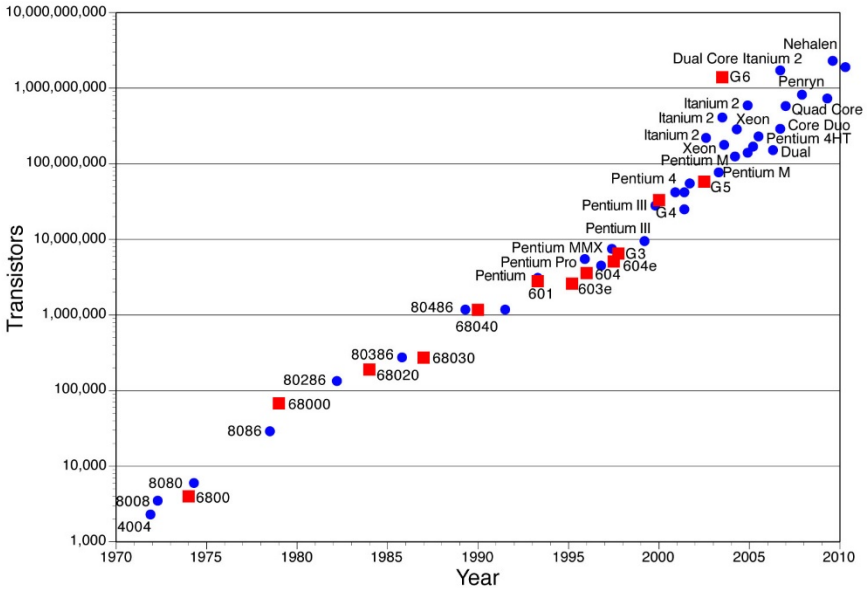
- Use ChangeCipherSpec message to start encrypting data
- Encryption: RC4, also DES, 3DES, AES, ...
- Authentication: HMAC, using MD5 or SHA1

SSL pitfalls

- Hard to set up
  - Expensive certificates
  - Resource-intensive
- Insufficient verification
  - Do people notice the lock icon?
  - Do people check the URL?
- Improper use

Forward Secrecy

- Under traditional HTTPS
  - The client chooses a random session key
  - Encrypt it using the server's RSA public key
  - Send **the encrypted session key over the network** to the server.
  - The server decrypts the session key with its RSA private key and use the decrypted session key to communicate with the client
- What if an adversary records all the encrypted traffic
  - And eventually, he breaks the server's RSA private key…



Forward Secrecy

- A session key derived from a set of long-term keys cannot be compromised if one of the long-term keys is compromised in the future
- How to achieve this?
  - Use Diffie-Hellman to establish session key
    - Session key is not transmitted on the network
  - Server's private key is only used to sign (authenticate) the key exchange to prevent man-in-the-middle attacks
    - Note that Diffie-Hellman works only for **authenticated** non-private channel.

Secure Shell

- SSH: secure shell
- Designed in 1995 by Tatu Ylonen
- Replaced in 1996 by SSHv2
  - Fixed security holes
  - Eventually standardized
- Less popular than SSL
  - But completely dominates the market

SSH architecture

- Similar to SSL:
  - Clients, servers, socket-like interface
- Difference:
  - No certificates
  - Remember public key associated with host
- Intuition
  - MITM attacks are difficult
  - To evade detection, adversary must succeed at MITM *every time*

Other SSH features

- Flexible authentication architecture
  - Password, public key, SecureID, Kerberos, ...
- Perfect forward secrecy
  - Permanent host key + temporary key
  - Host key signs temporary key
  - Key exchange done with temporary key
- Port forwarding
  - Used to tunnel traffic on other ports
  - Esp. X11 forwarding

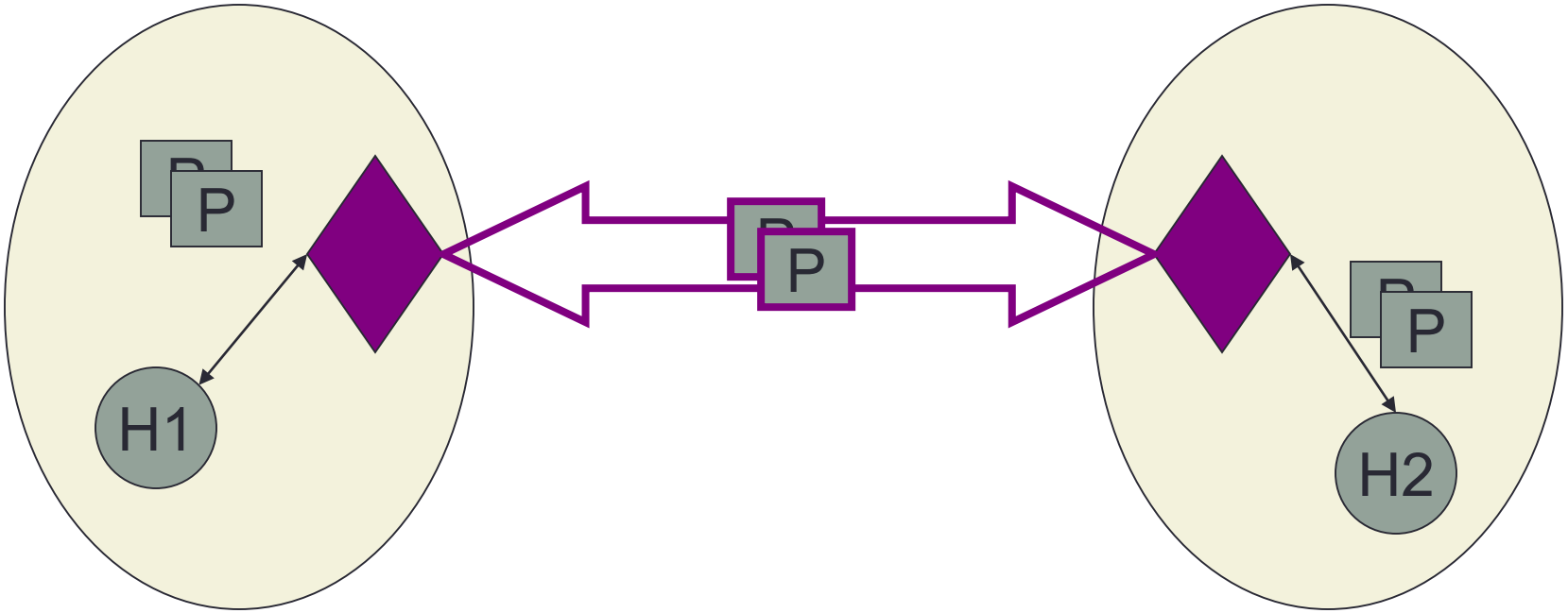
IPsec

- Designed as part of IPv6 suite
  - One of the key features v6 was supposed to bring
- Backported to IPv4
- Two options: AH (authentication) and ESP (encapsulated security)
- Two modes: transport and tunnel

Transport vs. Tunnel Mode

- Grand vision: eventually, all IP packets will be encrypted and authenticated
- Transport mode: add headers to IP to do so
- Reality: Most computers don't support IPsec
- Tunnel mode: use IPsec between two gateways to relay IP packets through "untrusted cloud"

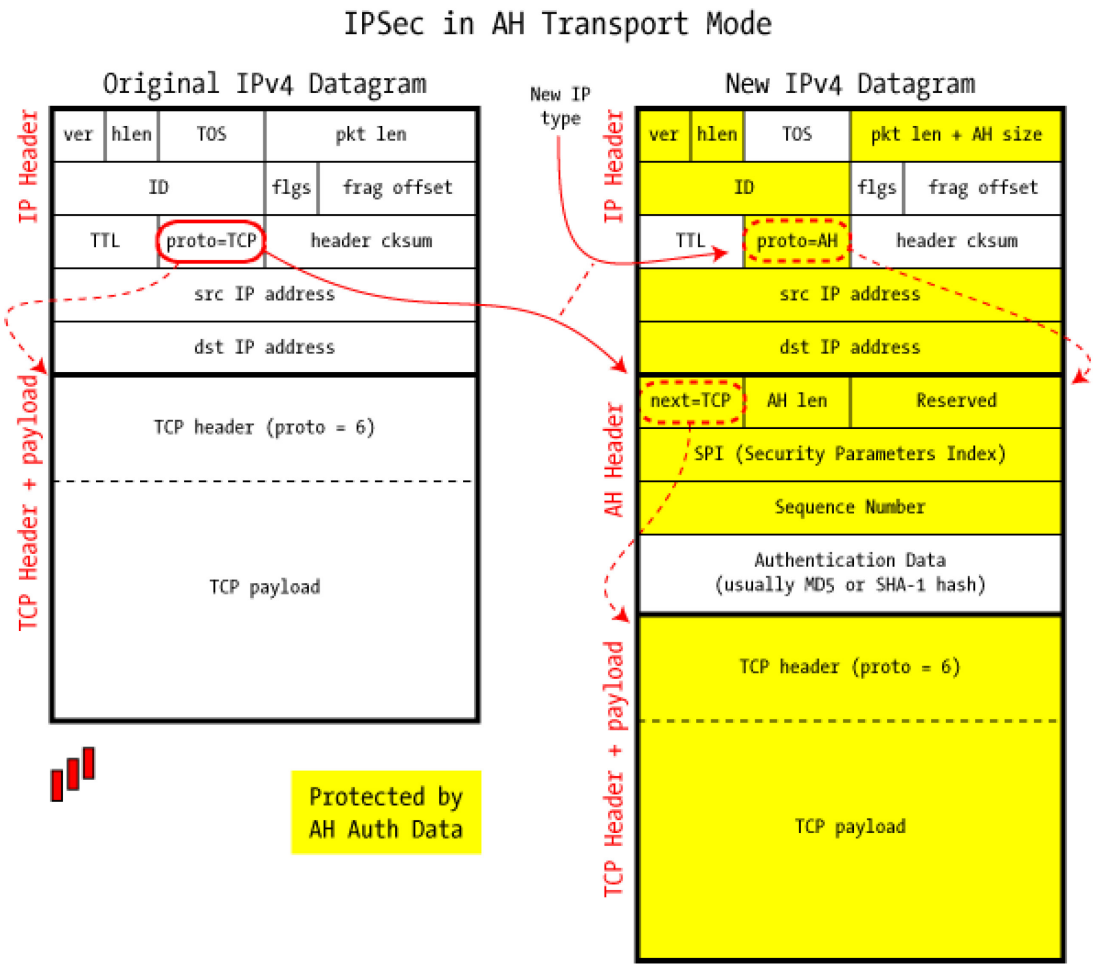
Tunnel Mode



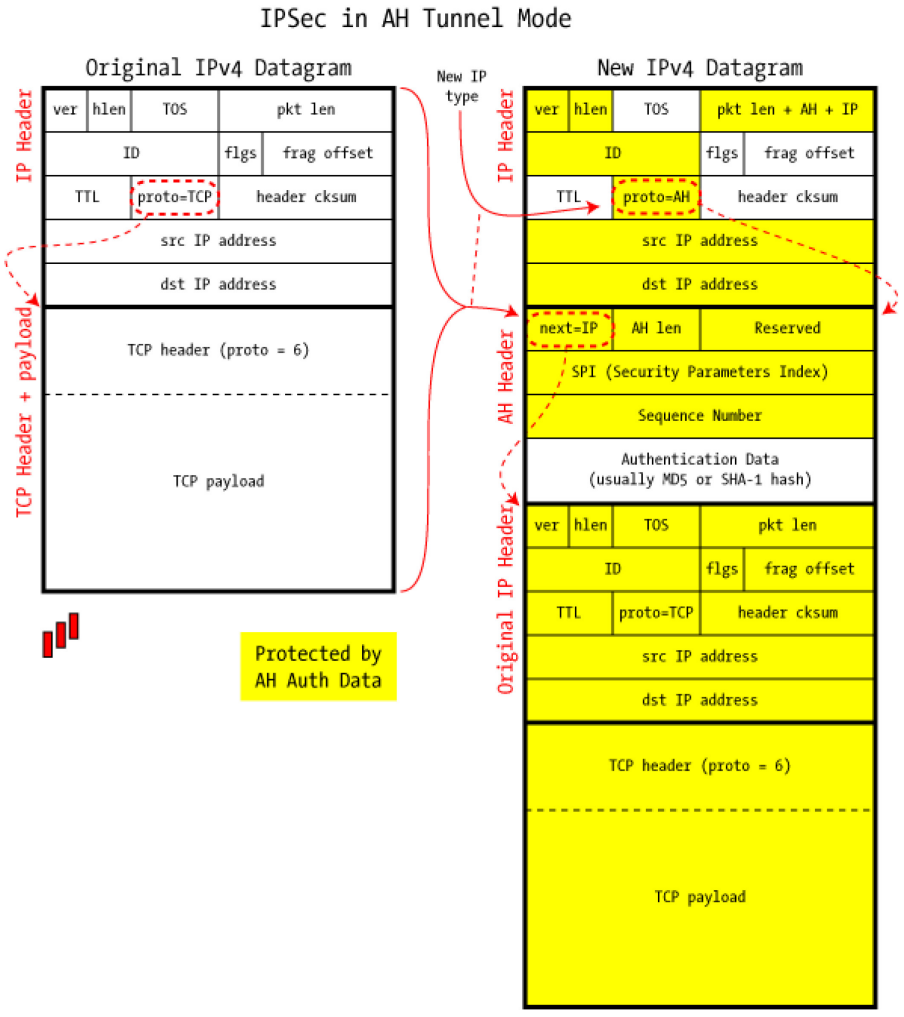
AH - Authentication

- Simple design: add header with authentication data
  - Security parameters
  - Authentication data (usu. SHA1-HMAC)

AH diagram


IPSec in AH Transport Mode
Original IPv4 Datagram
New IPv4 Datagram
New IP type
IP Header
TCP Header + payload
AH Header
ver
hlen
TOS
pkt len
pkt len + AH size
ID
flgs
frag offset
TTL
proto=TCP
proto=AH
header cksum
src IP address
dst IP address
next=TCP
AH len
Reserved
SPI (Security Parameters Index)
Sequence Number
Authentication Data
(usually MD5 or SHA-1 hash)
TCP header (proto = 6)
TCP payload
Protected by
AH Auth Data

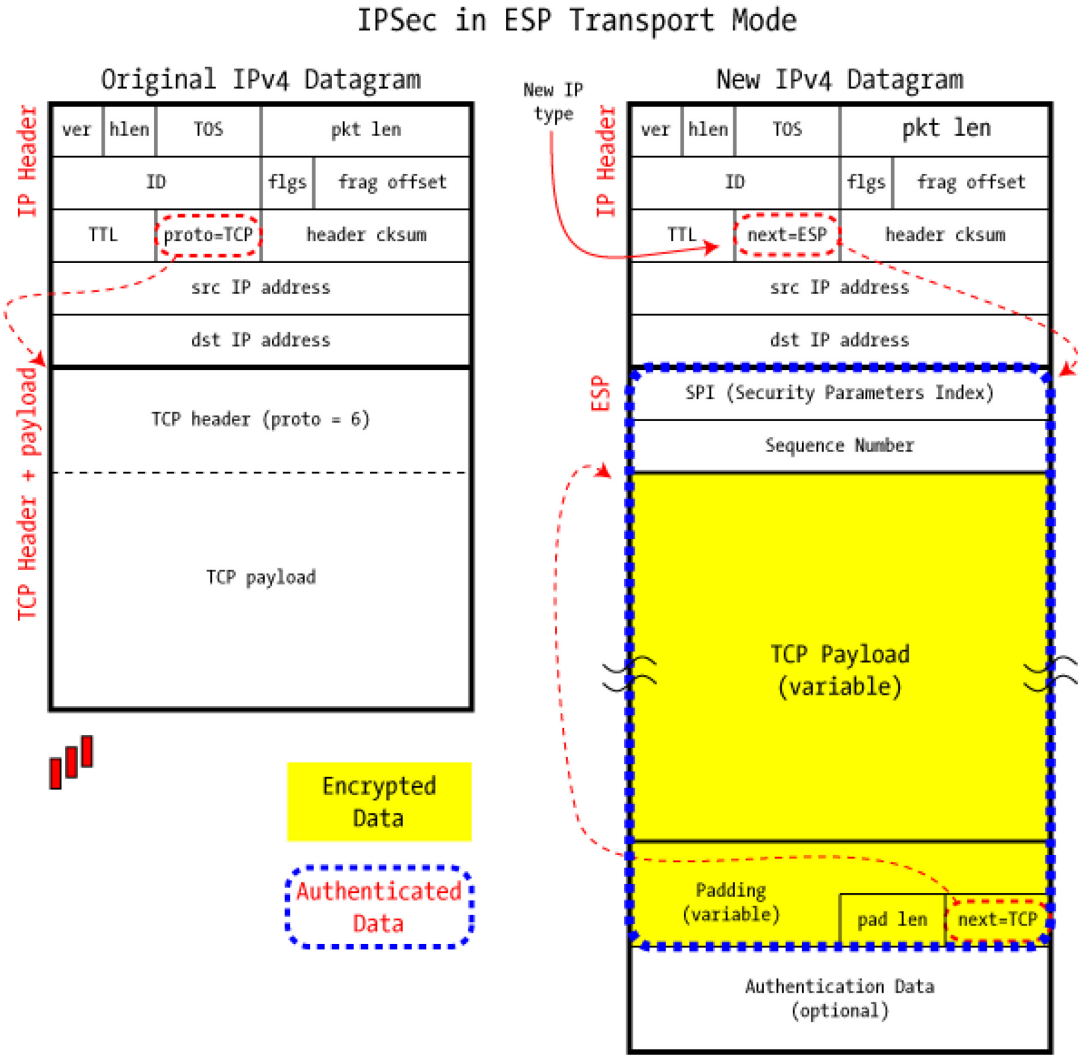
AH in Tunnel Mode



ESP - Encapsulated Security Payload

- Encapsulate data
  - Encapsulate datagram rather than add a header
  - Encrypt & authenticate

ESP diagram


IPSec in ESP Transport Mode
Original IPv4 Datagram
IP Header
ver
hlen
TOS
pkt len
ID
flgs
frag offset
TTL
proto=TCP
header cksum
src IP address
dst IP address
TCP Header + payload
TCP header (proto = 6)
TCP payload
Encrypted Data
Authenticated Data
New IPv4 Datagram
New IP type
IP Header
ver
hlen
TOS
pkt len
ID
flgs
frag offset
TTL
next=ESP
header cksum
src IP address
dst IP address
ESP
SPI (Security Parameters Index)
Sequence Number
TCP Payload (variable)
Padding (variable)
pad len
next=TCP
Authentication Data (optional)

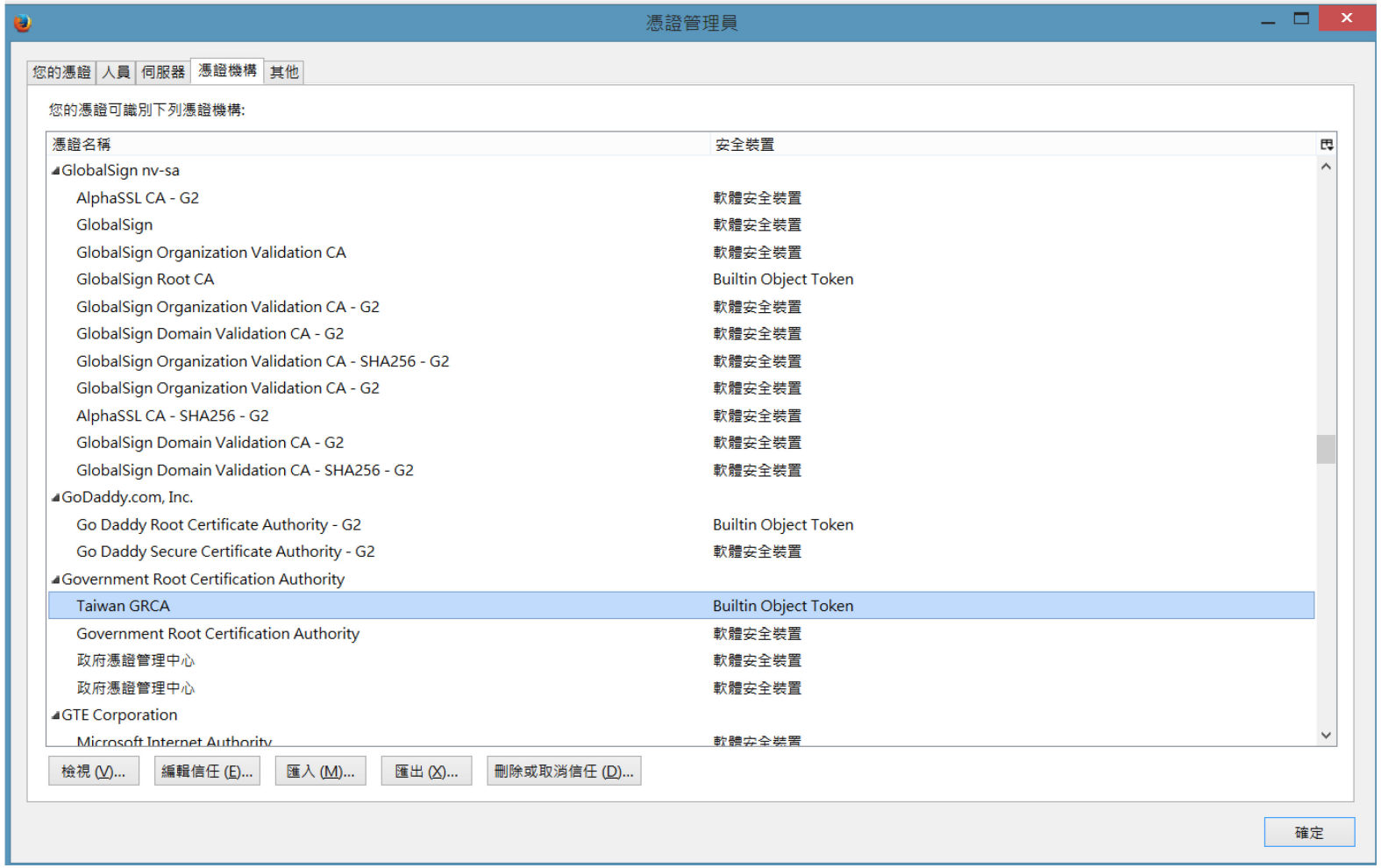
Key management

- ESP and AH use *session keys*
- Sessions are called *Security Associations*
  - Indexed by protocol, IP address, SPI
- ISAKMP: Internet Security Association Key Management Protocol
  - Authenticates parties
  - Establishes session keys
- Authentication
  - Big global PKI (DNSSEC??)
  - Manual configuration

IPsec redux

- Deployment of IPsec limited
- Some reasons
  - Global PKI infrastructure hard to set up
  - Wrong layer for security
    - Session/Application-layer, rather than Network layer
  - Fixes a “solved” problem
    - SSL & SSH work well
- IPsec success: VPNs
  - Use tunnel mode of Ipsec

PKI


憑證管理員
您的憑證
人員
伺服器
憑證機構
其他
您的憑證可識別下列憑證機構:
憑證名稱
安全裝置
GlobalSign nv-sa
AlphaSSL CA - G2
軟體安全裝置
GlobalSign
軟體安全裝置
GlobalSign Organization Validation CA
軟體安全裝置
GlobalSign Root CA
Builtin Object Token
GlobalSign Organization Validation CA - G2
軟體安全裝置
GlobalSign Domain Validation CA - G2
軟體安全裝置
GlobalSign Organization Validation CA - SHA256 - G2
軟體安全裝置
GlobalSign Organization Validation CA - G2
軟體安全裝置
AlphaSSL CA - SHA256 - G2
軟體安全裝置
GlobalSign Domain Validation CA - G2
軟體安全裝置
GlobalSign Domain Validation CA - SHA256 - G2
軟體安全裝置
GoDaddy.com, Inc.
Go Daddy Root Certificate Authority - G2
Builtin Object Token
Go Daddy Secure Certificate Authority - G2
軟體安全裝置
Government Root Certification Authority
Taiwan GRCA
Builtin Object Token
Government Root Certification Authority
軟體安全裝置
政府憑證管理中心
軟體安全裝置
政府憑證管理中心
軟體安全裝置
GTE Corporation
檢視 (V)...
編輯信任 (E)...
匯入 (M)...
匯出 (X)...
刪除或取消信任 (D)...
確定

Recent problems with SSL

- Issues with the security of Certificate Authorities
  - Comodo, Diginotar, KPN, Trustwave, …
- News on international espionage
  - Attacks against CAs
  - Compelled certificate attack (i.e. a government orders a CA to issue a false certificate)

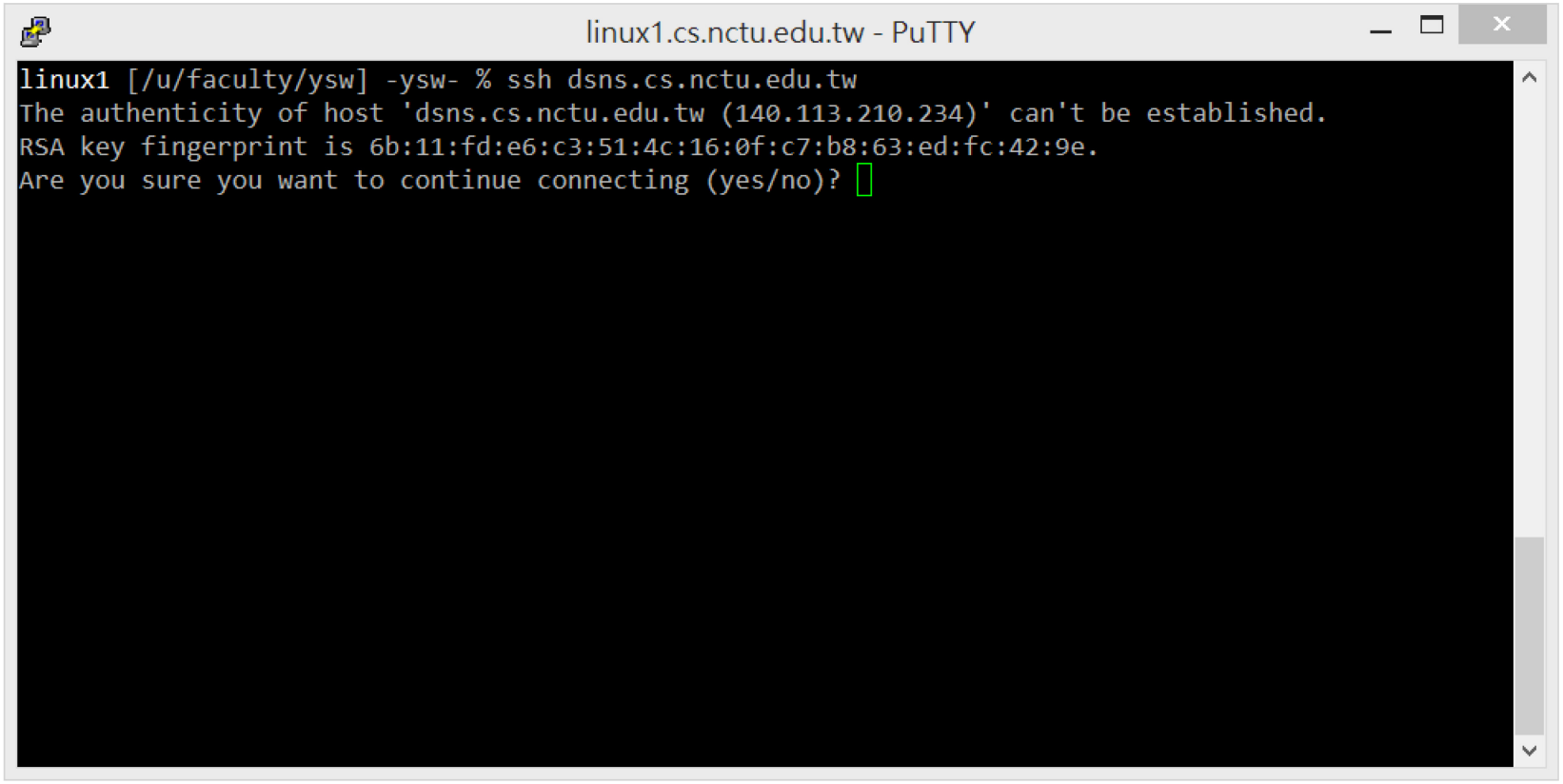
Recent problems with SSL

- Weaknesses in the protocol
  - renegotiation, BEAST, CRIME, etc.
- Weaknesses in SSL implementations
  - gotofail, heartbleed, change cipher spec (CCS) injection, etc.
- Weak SSL keys in large numbers (0.2% of all keys on the web)

Issues of Certificate Authorities

- There are about 100+ CAs globally; Browsers trust all of them
  - If one CA is compromised, the attacker can impersonate any website
- Just having a few CAs is problematic
  - CAs operate in different countries and jurisdictions
    - They don't really trust each other
  - May work for certain applications
    - E.g. LINE,…
- Self-signed certificates
  - Man-in-the-middle attacks
- Self-signed certificates over a secure channel
  - Does not scale

Trust on First Use



Certificate Patrol

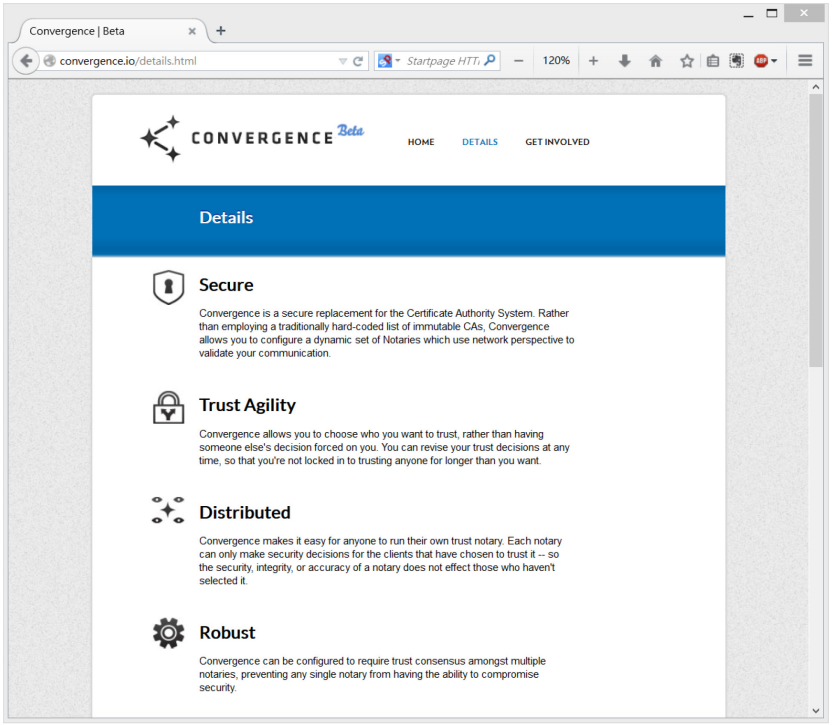
- Implement certificate pinning

- Same certificate?
- Same key?
- Same CA?

- https://addons.mozilla.org/en-US/firefox/addon/certificate-patrol/

Networked verification of identify

- Assumption: attacker may attack a single client but not the whole web
- http://convergence.io/




www.ithome.com.tw/news/91198
juiker ssl
iThome 新聞 產品評測 CIO 技術 專題 專欄 主題頻道 研討會 社群
資安
工研院研發通訊App揪科 證實有中間人攻擊風險
工研院研發手機通訊App揪科，日前被發現一個資安漏洞，證實可遭到中間人攻擊導致敏感個資外洩。工研院已經強化App端憑證安全性，新版App並於即日重新上架
文/ 黃彥棻 | 2014-10-01 發表
讚 7,884 按讚加入iThome粉絲團
讚 分享 586
+1 27
Juiker
服務特色 會員權益 企業服務 說明支援 下載專區
網頁版通訊 企業會員登入
全新一代通訊軟
結合娛樂與商業用途，免費


web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-2014-6693
juiker ssl
120%
Sponsored by DHS National Cyber Security Division/US-CERT
NIST National Institute of Standards and Technology
National Vulnerability Database
automating vulnerability management, security measurement, and compliance checking
Vulnerabilities Checklists 800-53/800-53A Product Dictionary Impact Metrics Data Feeds Statistics FAQs
Home SCAP SCAP Validated Tools SCAP Events About Contact Vendor Comments
Mission and Overview
NVD is the U.S. government repository of standards based vulnerability management data. This data enables automation of vulnerability management, security measurement, and compliance (e.g. FISMA).
Resource Status
NVD contains:
65563 CVE Vulnerabilities
253 Checklists
248 US-CERT Alerts
3717 US-CERT Vuln Notes
10286 OVAL Queries
97219 CPE Names
Last updated: 10/15/2014 10:43:17 AM
CVE Publication rate: 32.77
Email List
NVD provides four mailing lists to the public. For information and subscription instructions
National Cyber Awareness System
Vulnerability Summary for CVE-2014-6693
Original release date: 09/23/2014
Last revised: 10/04/2014
Source: US-CERT/NIST
Overview
The Juiker (aka org.itri) application 3.2.0829.1 for Android does not verify X.509 certificates from SSL servers, which allows man-in-the-middle attackers to spoof servers and obtain sensitive information via a crafted certificate.
Impact
CVSS Severity (version 2.0):
CVSS v2 Base Score: 5.4 (MEDIUM) (AV:A/AC:M/Au:N/C:P/I:P/A:P) (legend)
Impact Subscore: 6.4
Exploitability Subscore: 5.5
CVSS Version 2 Metrics:
Access Vector: Local network exploitable; Victim must voluntarily interact with attack mechanism
Access Complexity: Medium
Authentication: Not required to exploit
Impact Type: Allows unauthorized disclosure of information; Allows unauthorized modification; Allows disruption of service
References to Advisories, Solutions, and Tools
By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest