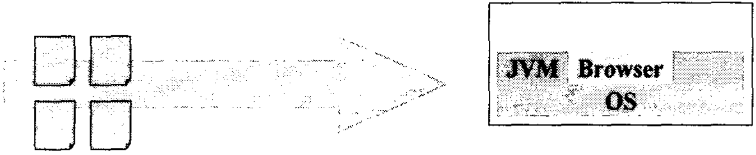
Security & Risks of Software Distribution in DTV Platforms

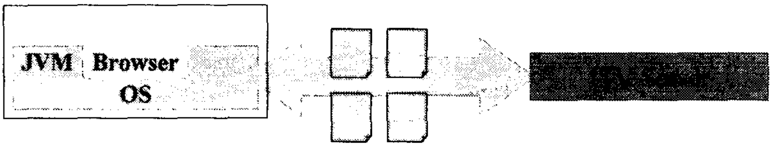
Edwin Heredia, PhD
Technical Manager
DTV Standards & Strategy

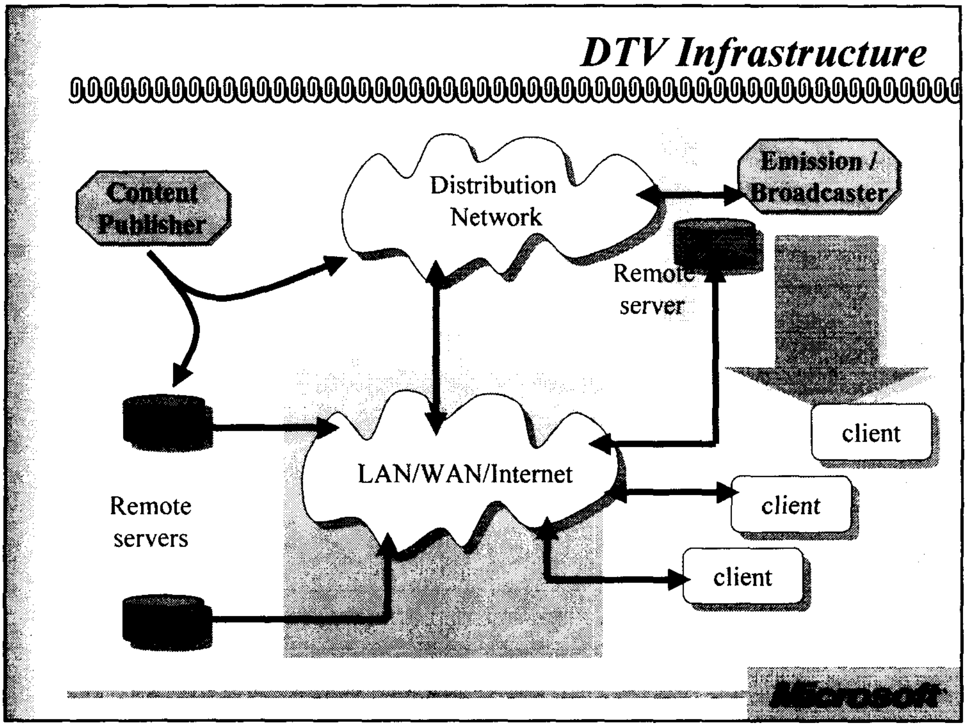
Software Distribution in the TV Space

DASE, MHP, OCAP provide the means to broadcast software to millions of devices simultaneously.



DASE, MHP, OCAP provide the means for devices to exchange data with servers when an interactive channel is available.





DTV Infrastructure

Content Publisher

Distribution Network

Emission / Broadcaster

Remote server

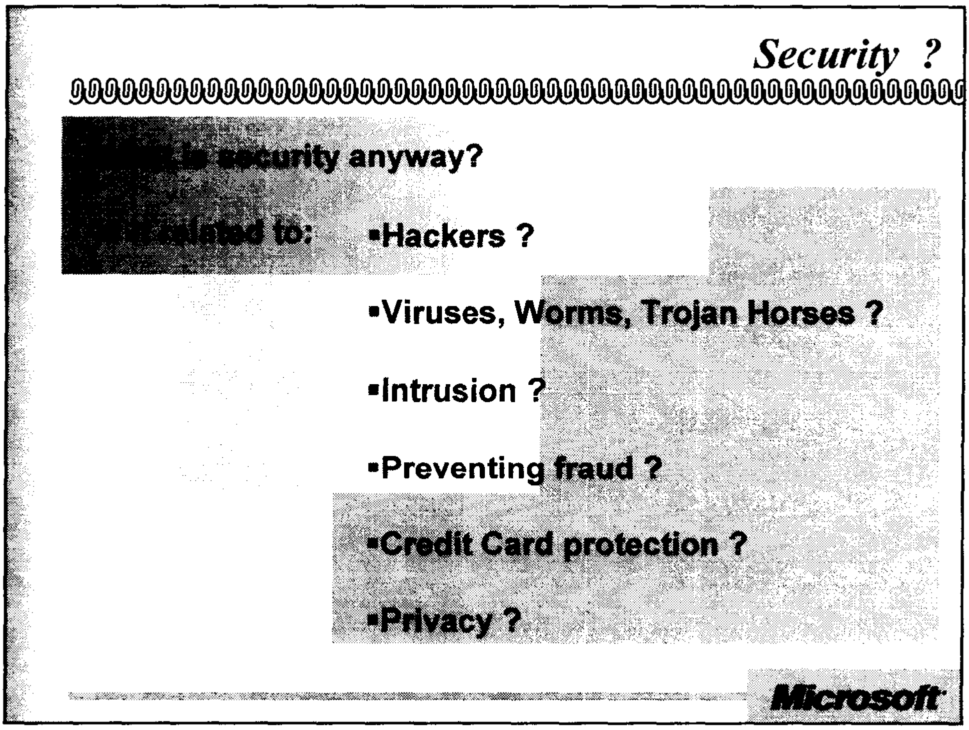
Remote servers

LAN/WAN/Internet

client

client

client

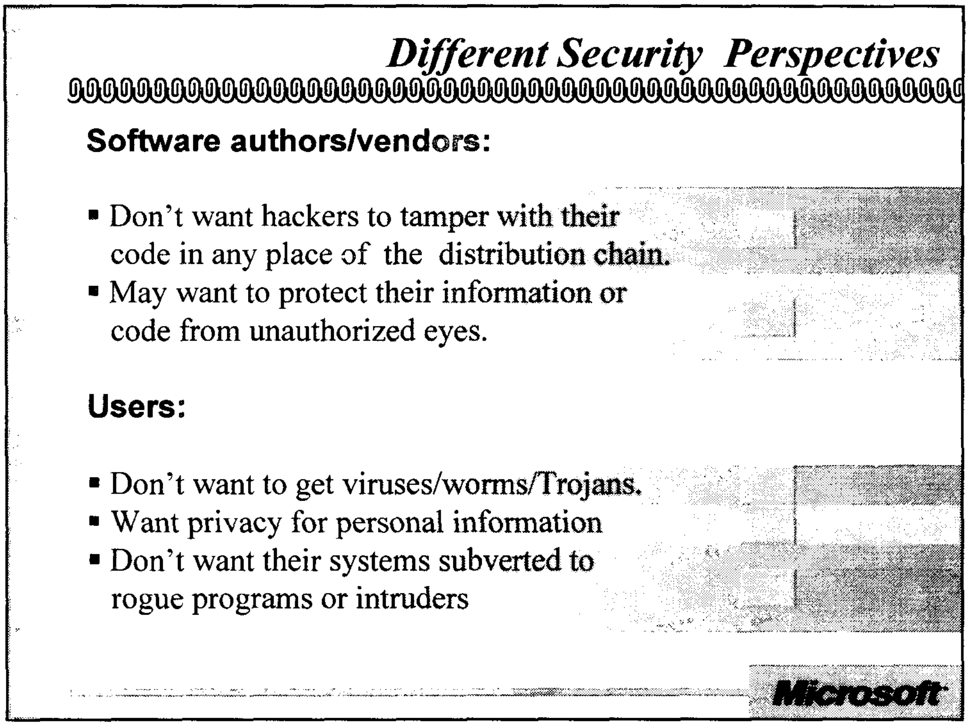


Security ?

[illegible] is security anyway?

[illegible] related to:

- Hackers ?
- Viruses, Worms, Trojan Horses ?
- Intrusion ?
- Preventing fraud ?
- Credit Card protection ?
- Privacy ?



Different Security Perspectives

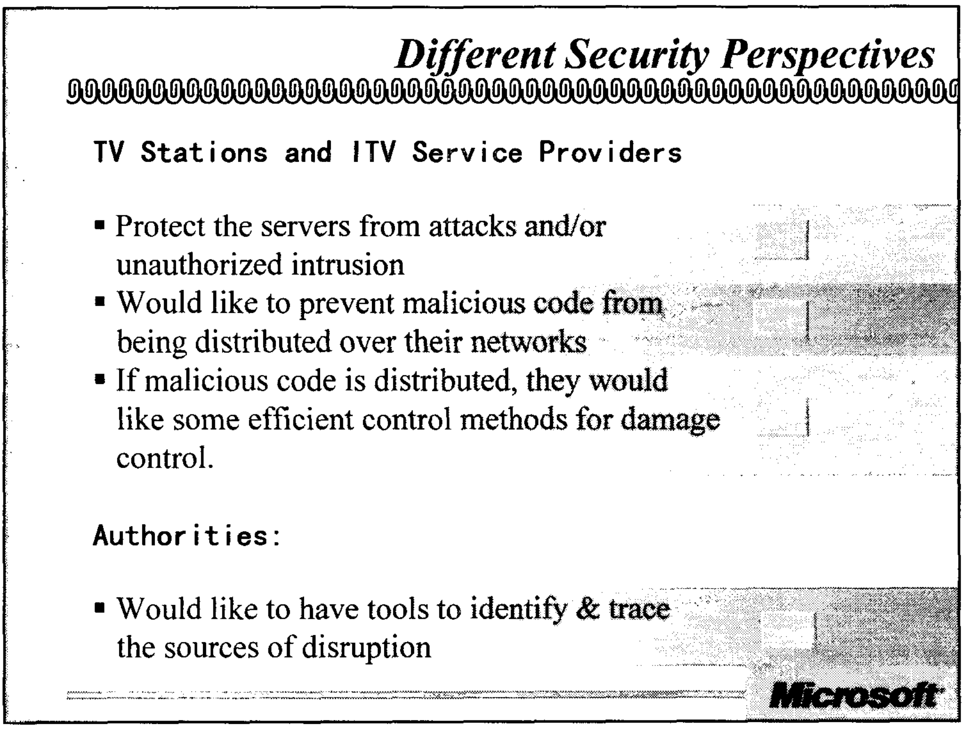
Software authors/vendors:

- Don’t want hackers to tamper with their code in any place of the distribution chain.
- May want to protect their information or code from unauthorized eyes.

Users:

- Don’t want to get viruses/worms/Trojans.
- Want privacy for personal information
- Don’t want their systems subverted to rogue programs or intruders

Microsoft



Different Security Perspectives

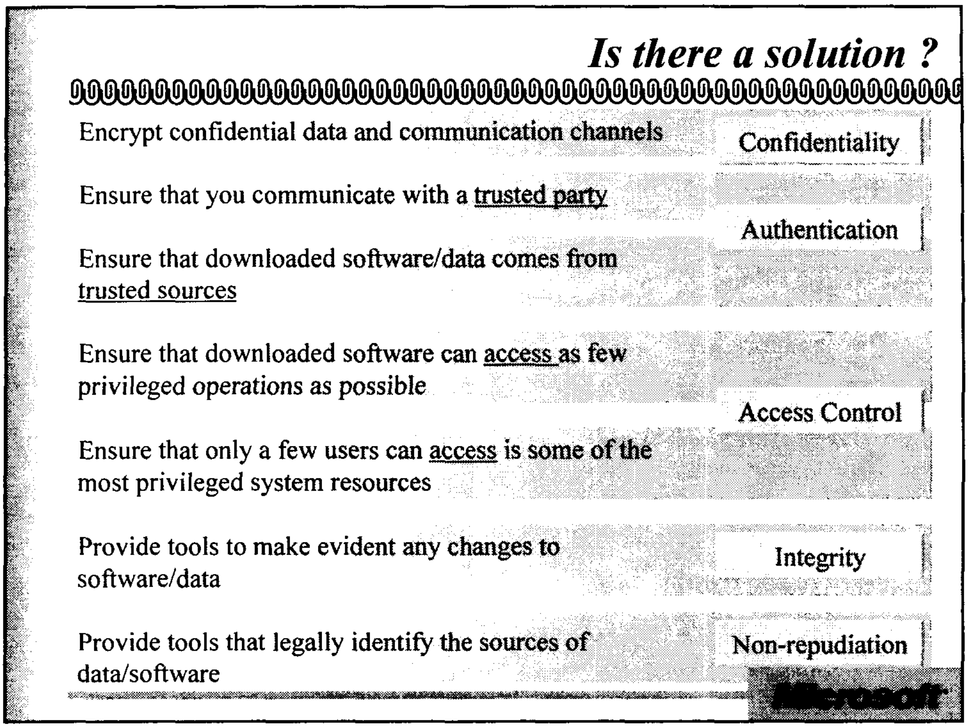
TV Stations and ITV Service Providers

- Protect the servers from attacks and/or unauthorized intrusion
- Would like to prevent malicious code from being distributed over their networks
- If malicious code is distributed, they would like some efficient control methods for damage control.

Authorities:

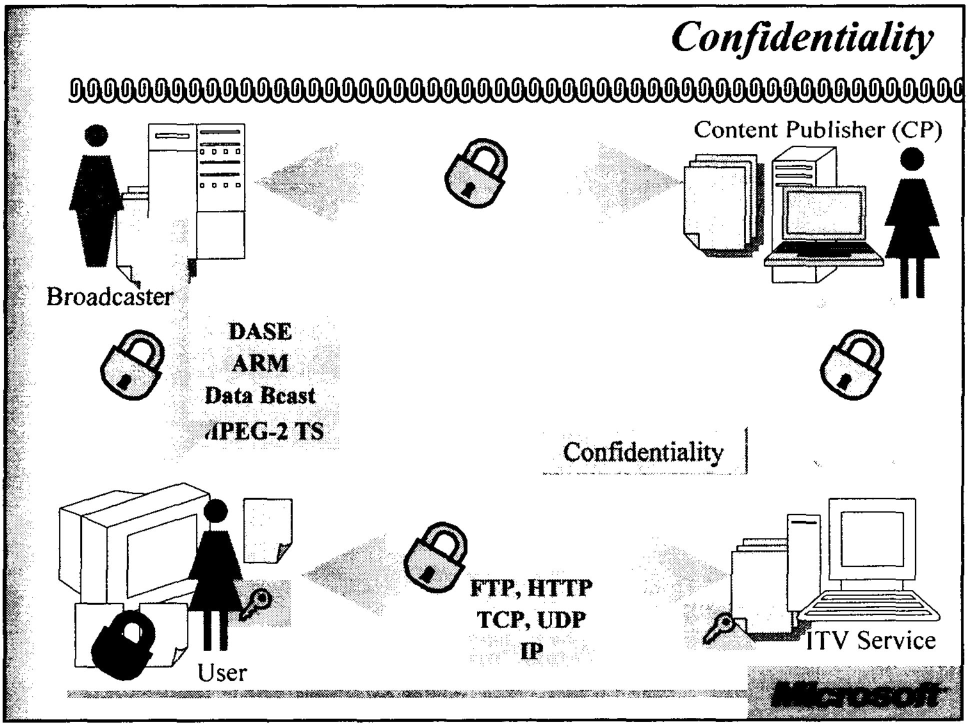
- Would like to have tools to identify & trace the sources of disruption

Microsoft



Is there a solution ?

Encrypt confidential data and communication channels	Confidentiality
Ensure that you communicate with a trusted party	Authentication
Ensure that downloaded software/data comes from trusted sources	
Ensure that downloaded software can access as few privileged operations as possible	Access Control
Ensure that only a few users can access is some of the most privileged system resources	
Provide tools to make evident any changes to software/data	Integrity
Provide tools that legally identify the sources of data/software	Non-repudiation



Confidentiality

Content Publisher (CP)

Broadcaster

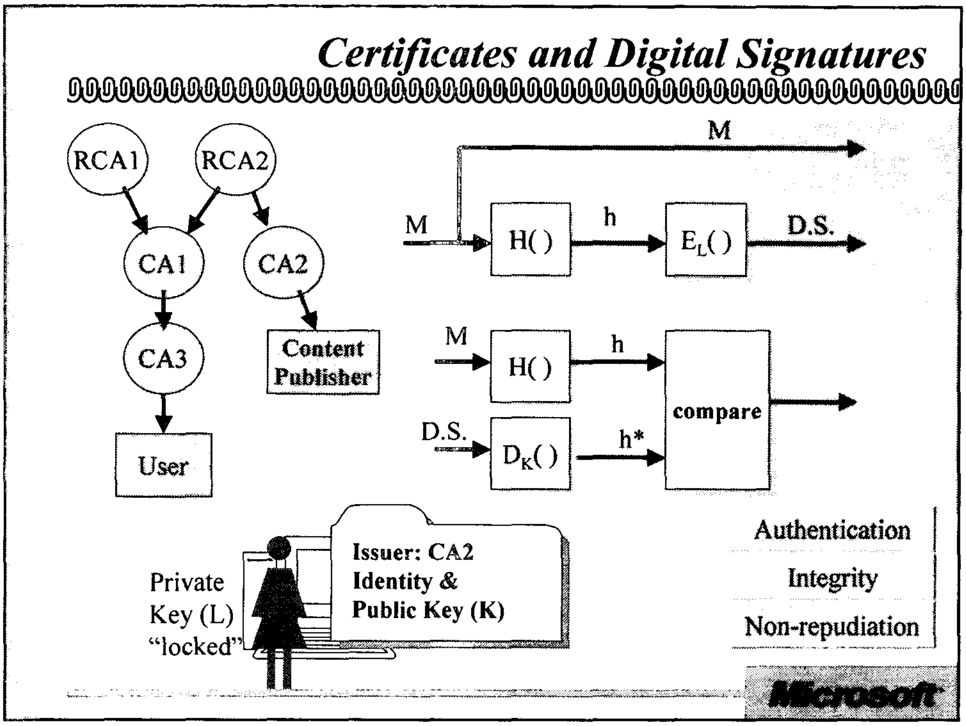
DASE
ARM
Data Bcast
ıPEG-2 TS

Confidentiality

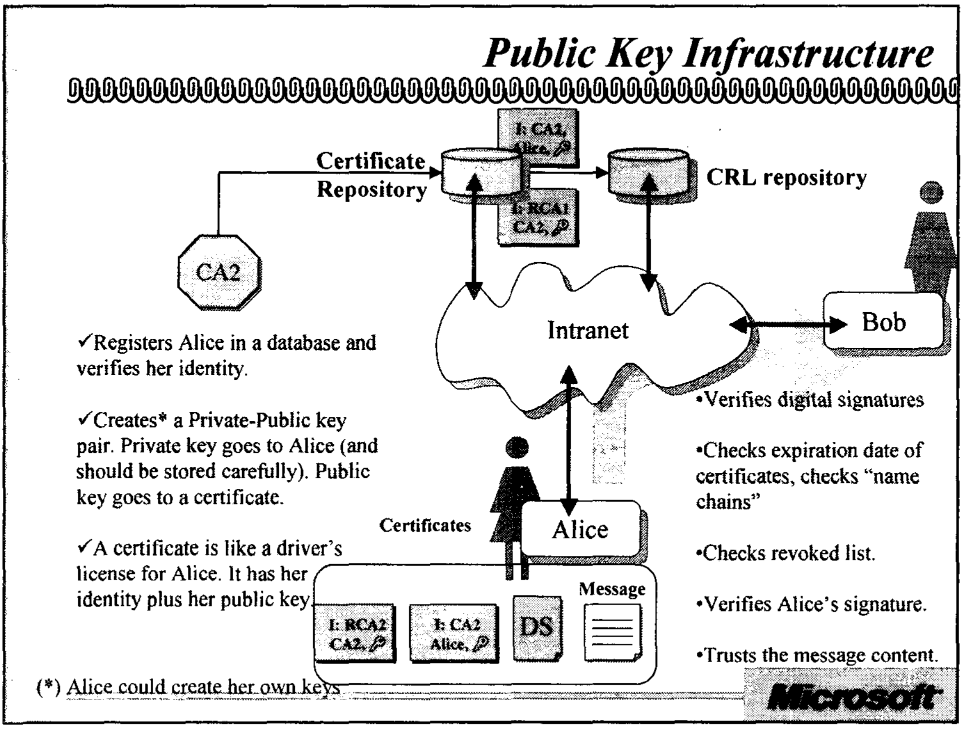
FTP, HTTP
TCP, UDP
IP

User

ITV Service



Certificates and Digital Signatures



Public Key Infrastructure

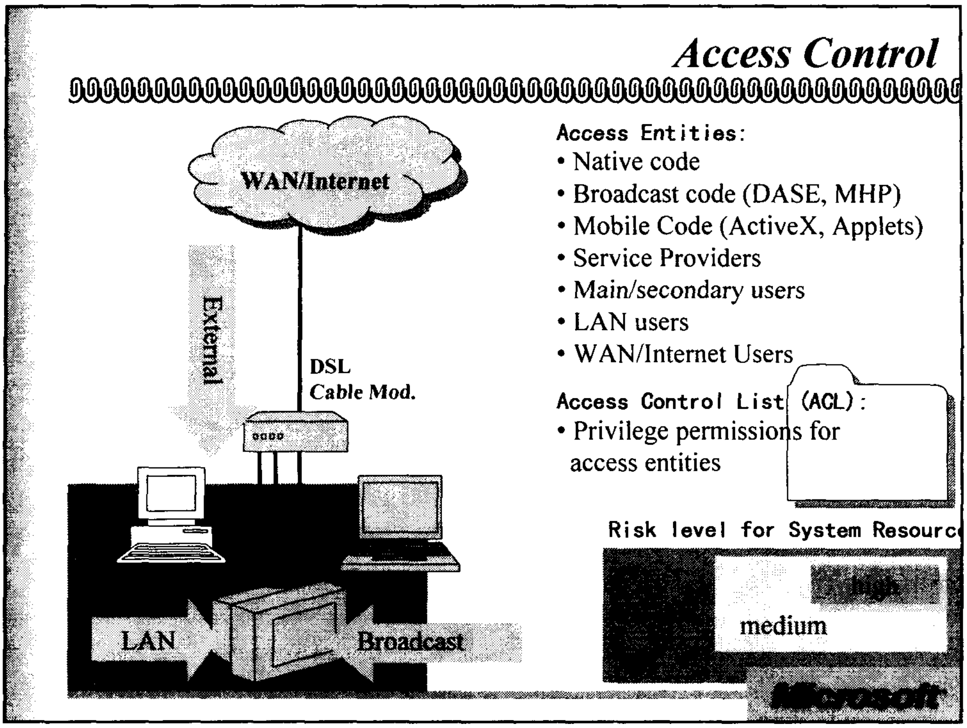
✓Registers Alice in a database and verifies her identity.

✓Creates* a Private-Public key pair. Private key goes to Alice (and should be stored carefully). Public key goes to a certificate.

✓A certificate is like a driver's license for Alice. It has her identity plus her public key.

- Verifies digital signatures
- Checks expiration date of certificates, checks "name chains"
- Checks revoked list.
- Verifies Alice's signature.
- Trusts the message content.

(*) Alice could create her own keys



Access Control

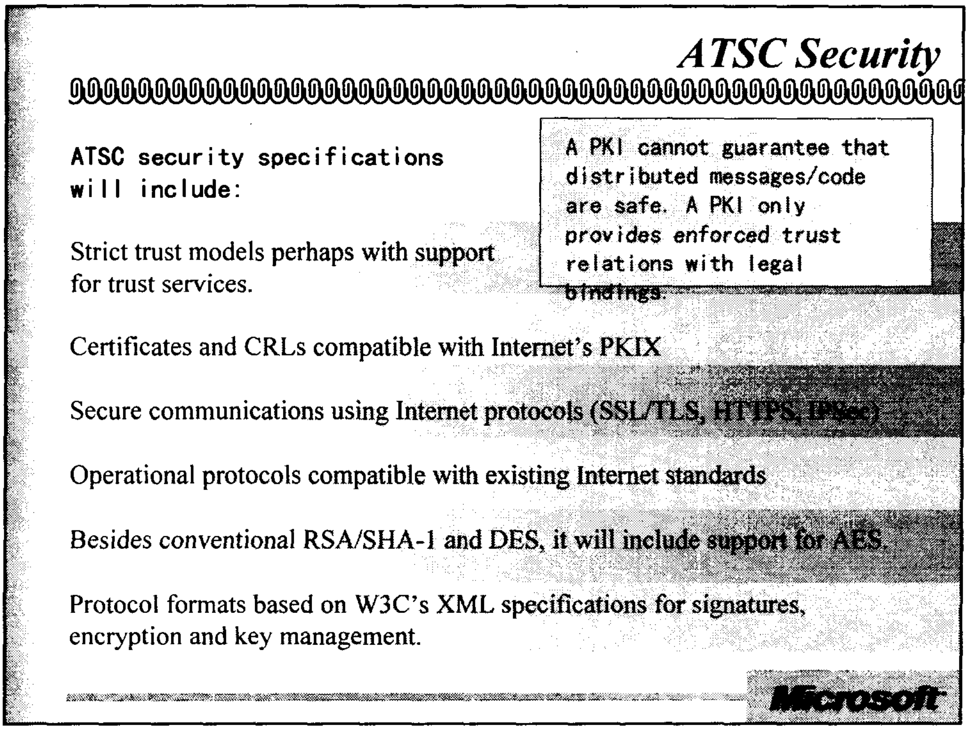
Access Entities:
- Native code
- Broadcast code (DASE, MHP)
- Mobile Code (ActiveX, Applets)
- Service Providers
- Main/secondary users
- LAN users
- WAN/Internet Users

Access Control List (ACL):
- Privilege permissions for access entities

Risk level for System Resourc

high

medium



ATSC Security

ATSC security specifications will include:

A PKI cannot guarantee that distributed messages/code are safe. A PKI only provides enforced trust relations with legal bindings.

Strict trust models perhaps with support for trust services.

Certificates and CRLs compatible with Internet's PKIX

Secure communications using Internet protocols (SSL/TLS, HTTPS, IPSec)

Operational protocols compatible with existing Internet standards

Besides conventional RSA/SHA-1 and DES, it will include support for AES.

Protocol formats based on W3C's XML specifications for signatures, encryption and key management.

Security Hypothesis

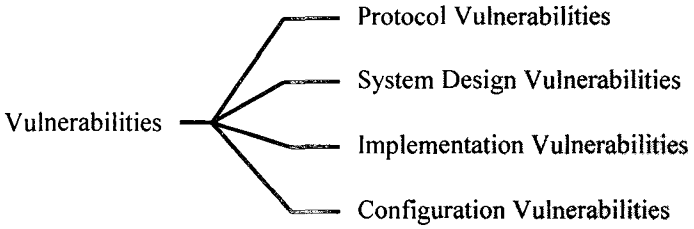
- "Only trusted applications will run privileged operations"
- "A PKI makes it difficult to insert malicious code into the system"
- "Encrypted channels will prevent fraud during e-commerce transactions"
- "Certificates provide legally binding identification"

Yes but...

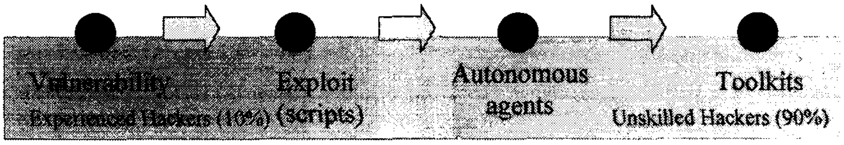
"The difficulty of securing a system is directly proportional to its complexity"

Microsoft

Security Vulnerabilities



Vulnerabilities
Protocol Vulnerabilities
System Design Vulnerabilities
Implementation Vulnerabilities
Configuration Vulnerabilities



Microsoft

Breaking the Security Hypothesis

CERT Vulnerability Notes (www.cert.org)

VU#32231 (08/03/2000): Netscape Communicator and Navigator ship with Java classes that allow an unsigned Java applet to access local and remote resources in violation of the security policies for applets.

VU#31607 (08/02/2000): A Vulnerability exists in the Microsoft Windows 2000 Service Control Manager which could allow local users to gain control of the system.

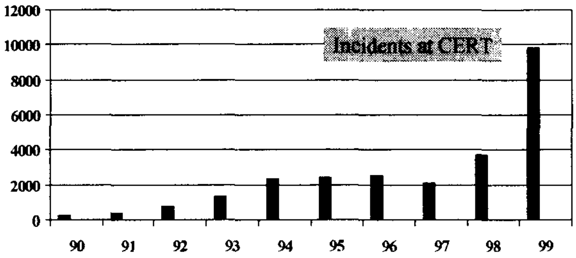
VU#25701 (07/27/2000) Linux gpm version 1.19.2 and earlier are vulnerable due to a flaw that allows a local user to delete arbitrary files.

VU#24346 (04/26/2000) Cisco IOS software allows an attacker to crash and reboot affected switches and routers. The problem occurs when the HTTP interface receives a request for "http://router-ip/anytext/%%"

Incidents and Attacks

Attack: Unauthorized access/use attempt regardless of success.

Incident: A group of attacks that have some clear distinctiveness.

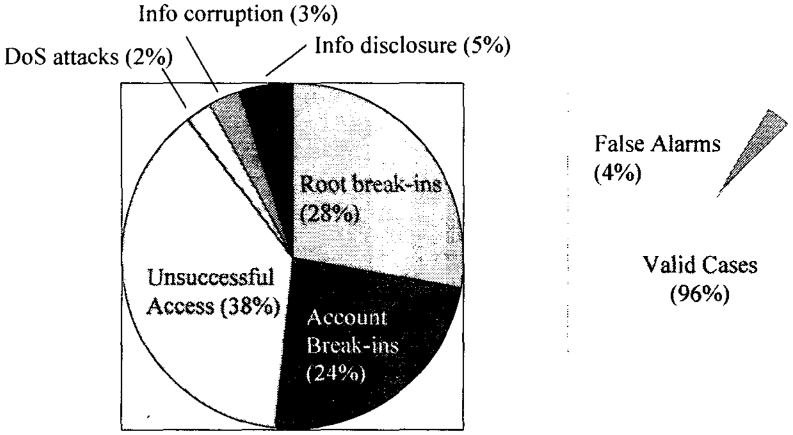


The estimated number of attacks in 1995 was from 40,000 to 2.5 million

The estimated number of incidents in 1995 was from 1,200 to 22,800

Source of statistics: John D. Howard, "An Analysis of Security Incidents on the Internet 1989-1995", PhD Thesis, Carnegie Mellon University, 1997.

Incident Distribution



Source of statistics: John D. Howard, "An Analysis of Security Incidents on the Internet 1989-1995", PhD Thesis, Carnegie Mellon University, 1997.

Microsoft

Corollary

Security is a shared responsibility between protocol/system design, implementation and the user.

Cryptography and integration protocols constitute only [illegible] story, the other half is in implementation.

Security should be a top priority of software development. Get all your developers thinking about security. Establish teams to review and audit the code frequently, and give security the highest priority during quality control.

After all, it is always much cheaper to fix security bugs [illegible] and not later once an attack has occurred.

Thanks !

Microsoft