Network Security Protocols: Analysis methods and standards

John Mitchell

Stanford University

Joint work with many students, postdocs, collaborators

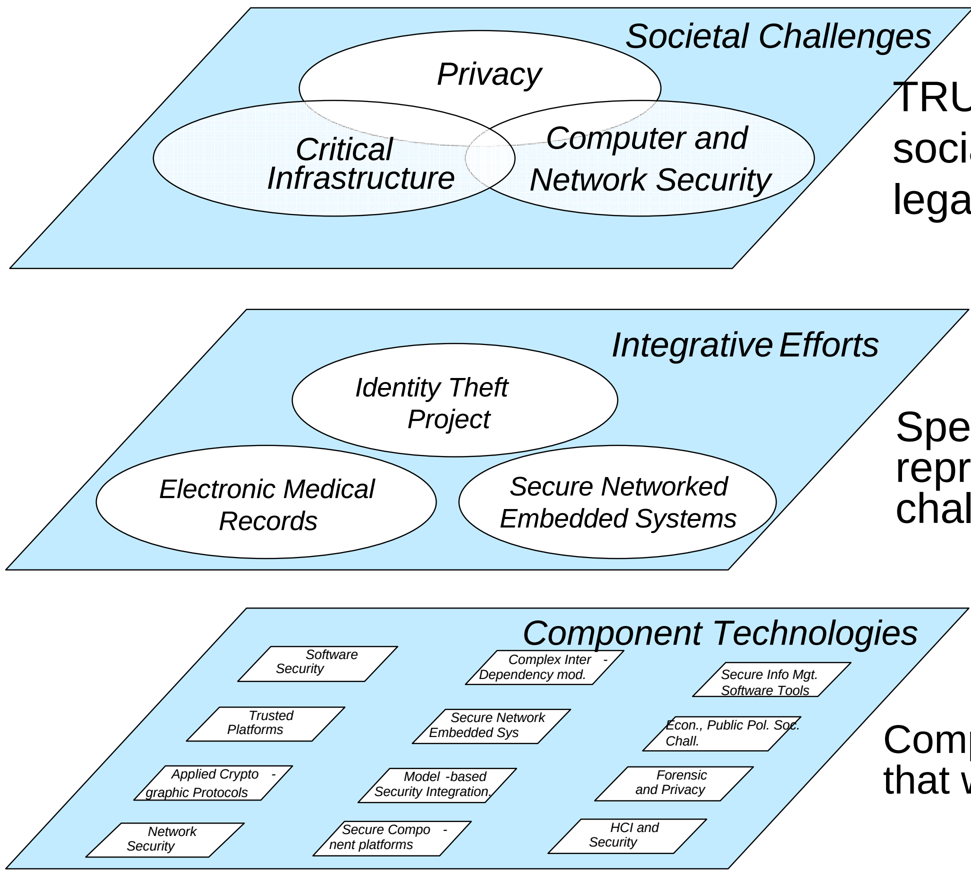
TRUST: Team for Research in Ubiquitous Secure Technologies

Carnegie Mellon · Cornell University · Mills College · San José State University · Smith College · Stanford University · Berkeley University of California · Vanderbilt University

NSF Science and Technology Center
Multi-university multi-year effort
Research, education, outreach

http://trust.eecs.berkeley.edu/

TRUST Research Vision



TRUST will address social, economic and legal challenges

Specific systems that represent these social challenges.

Component technologies that will provide solutions

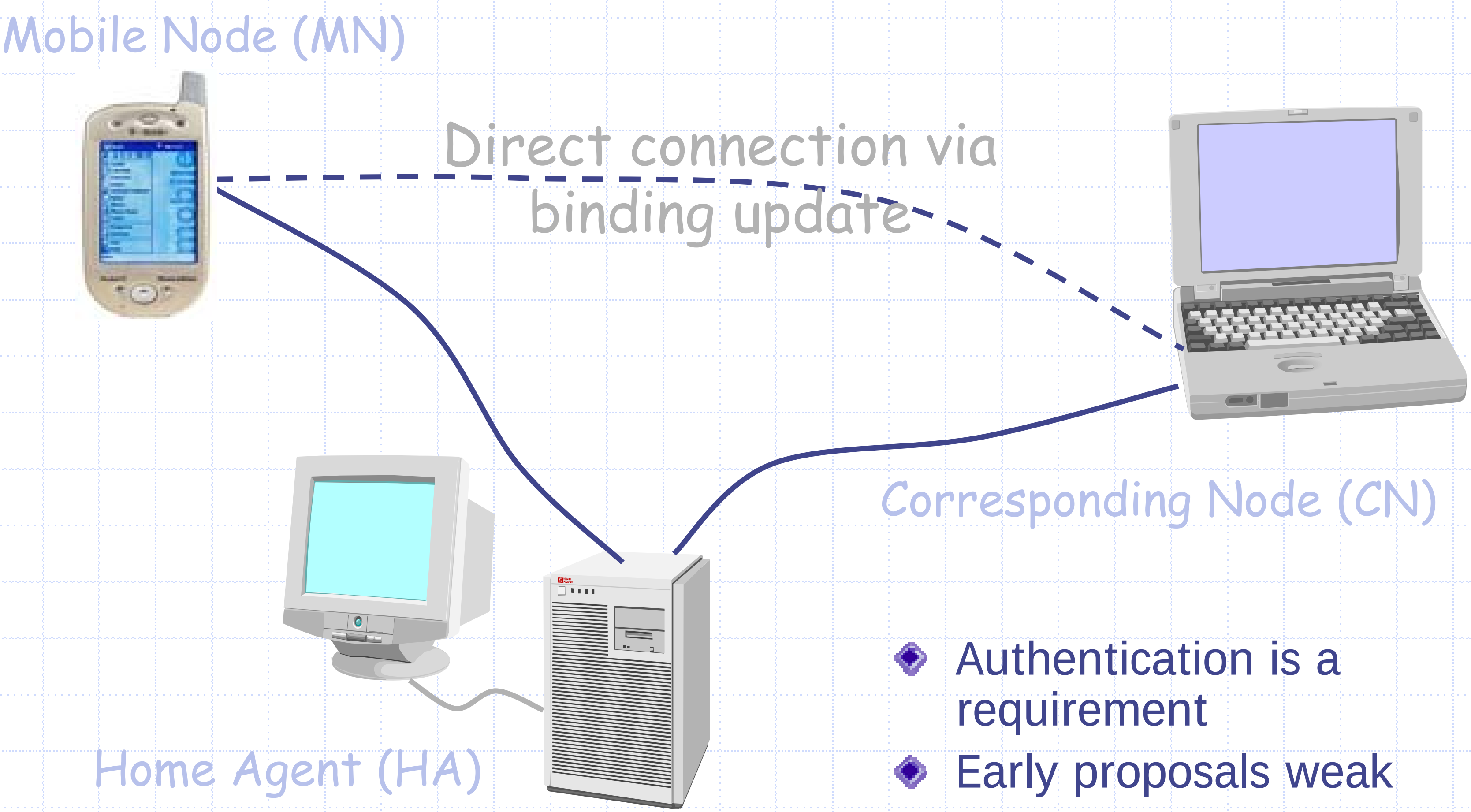
Network security protocols

- Primarily key management
  - Cryptography reduces many problems to key management
  - Also denial-of-service, other issues
- Hard to design and get right
  - People can do an acceptable job, eventually
  - Systematic methods improve results
- Practical case for software verification
  - Even for standards that are widely used and carefully reviewed, automated tools find flaws

Recent and ongoing protocol efforts

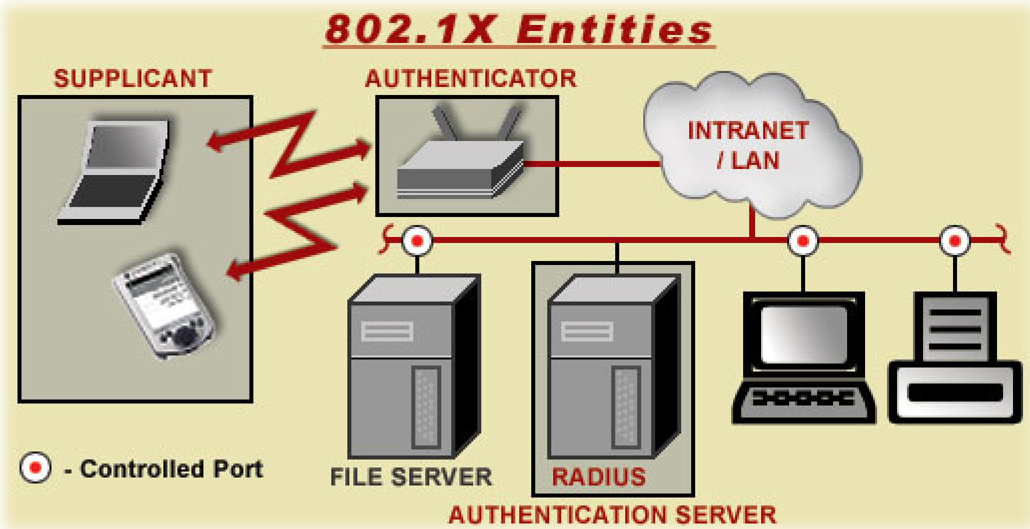
- Wireless networking authentication
  - 802.11i – improved auth for access point
  - 802.16e – metropolitan area networks
  - Simple config – setting up access point
- Mobility
  - Mobile IPv6 – update IP addr to avoid triangle routing
- VoIP
  - SIP – call referral feature, other issues
- Kerberos
  - PKINIT – public-key method for cross-domain authentication
- IPSec
  - IKEv1, JFK, IKEv2 – improved key management

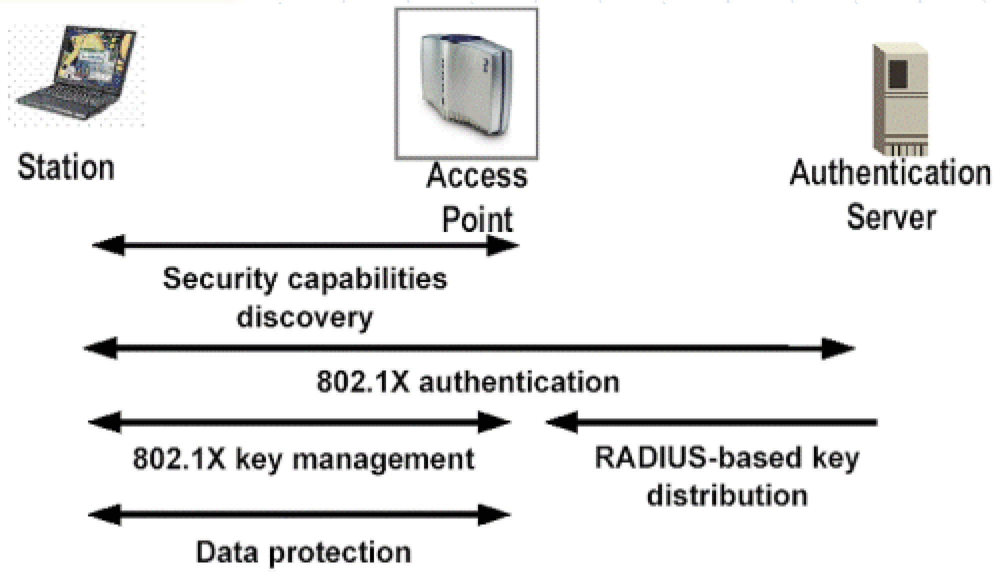
Mobile IPv6 Architecture



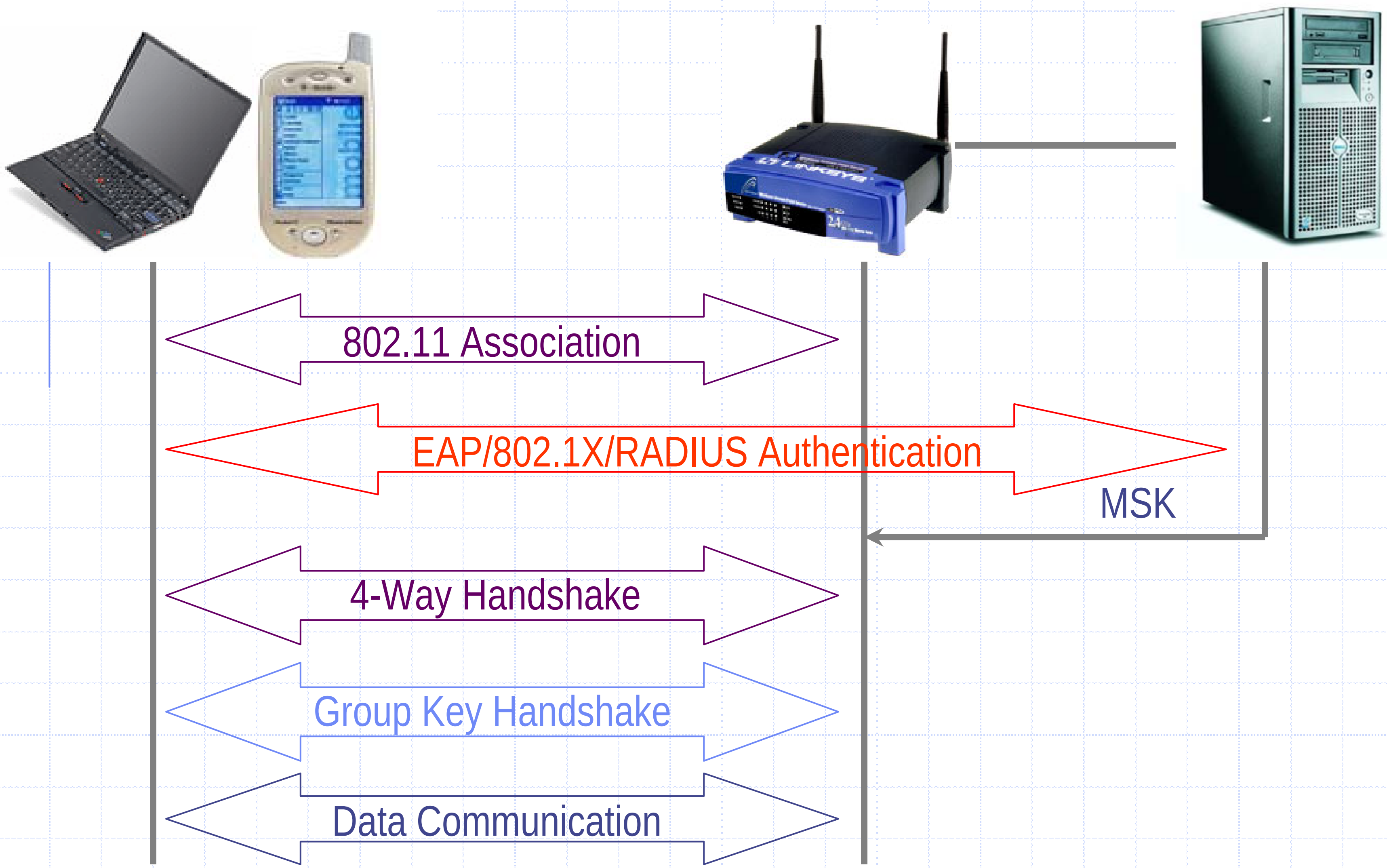
- Authentication is a requirement
- Early proposals weak

Wireless Authentication

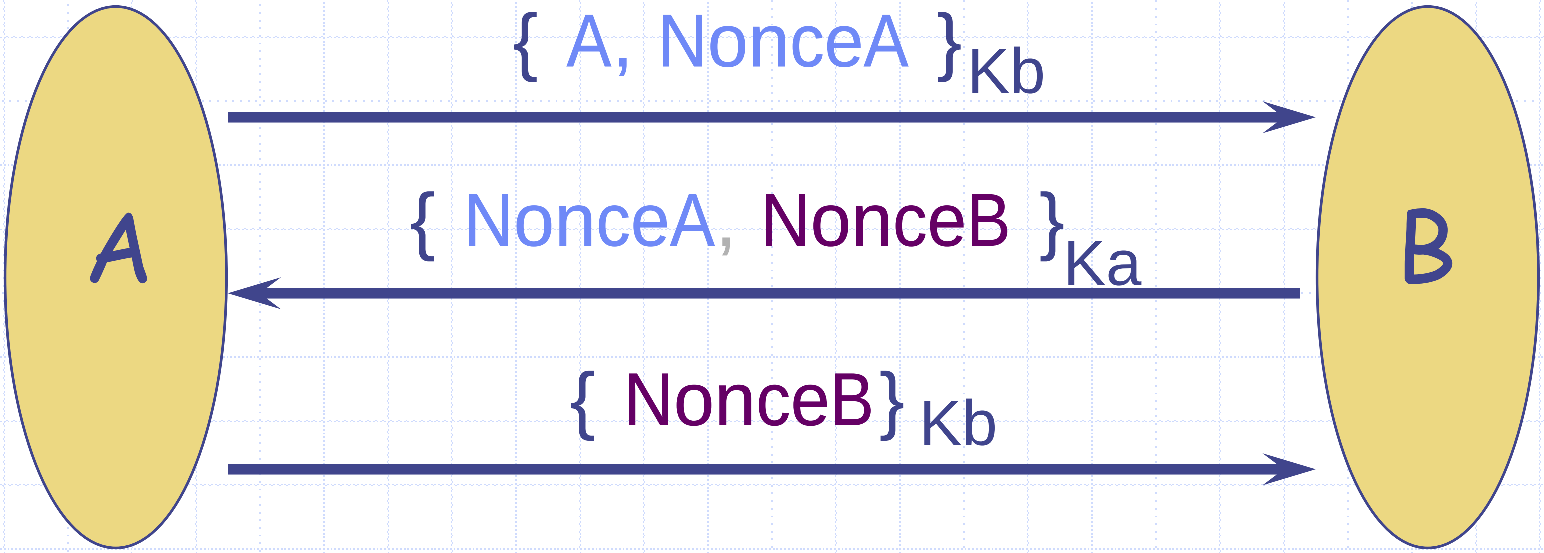




802.11i Protocol



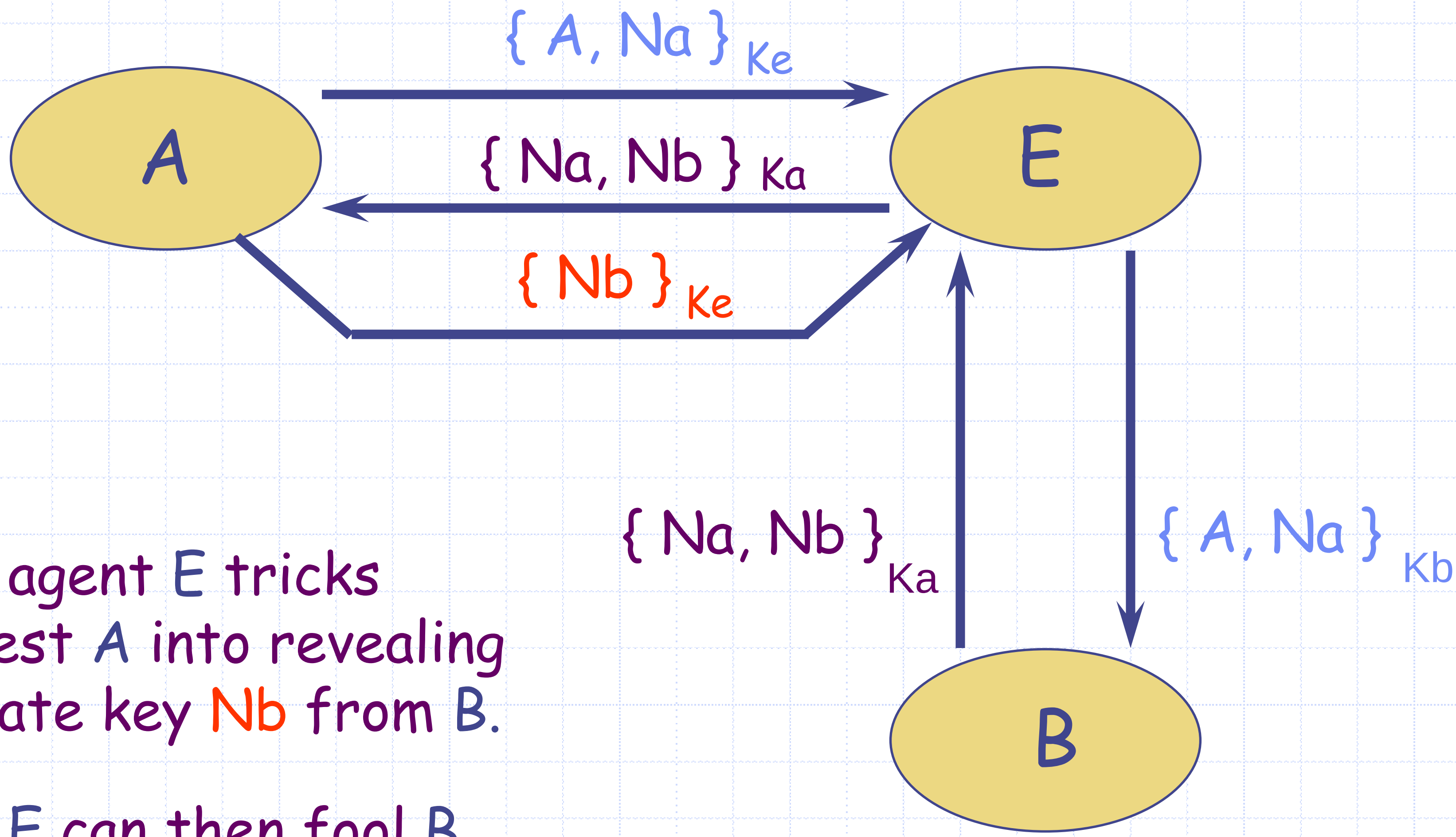
Needham-Schroeder Protocol



Result: A and B share two private numbers not known to any observer without Ka^{-1}, Kb^{-1}

Anomaly in Needham-Schroeder

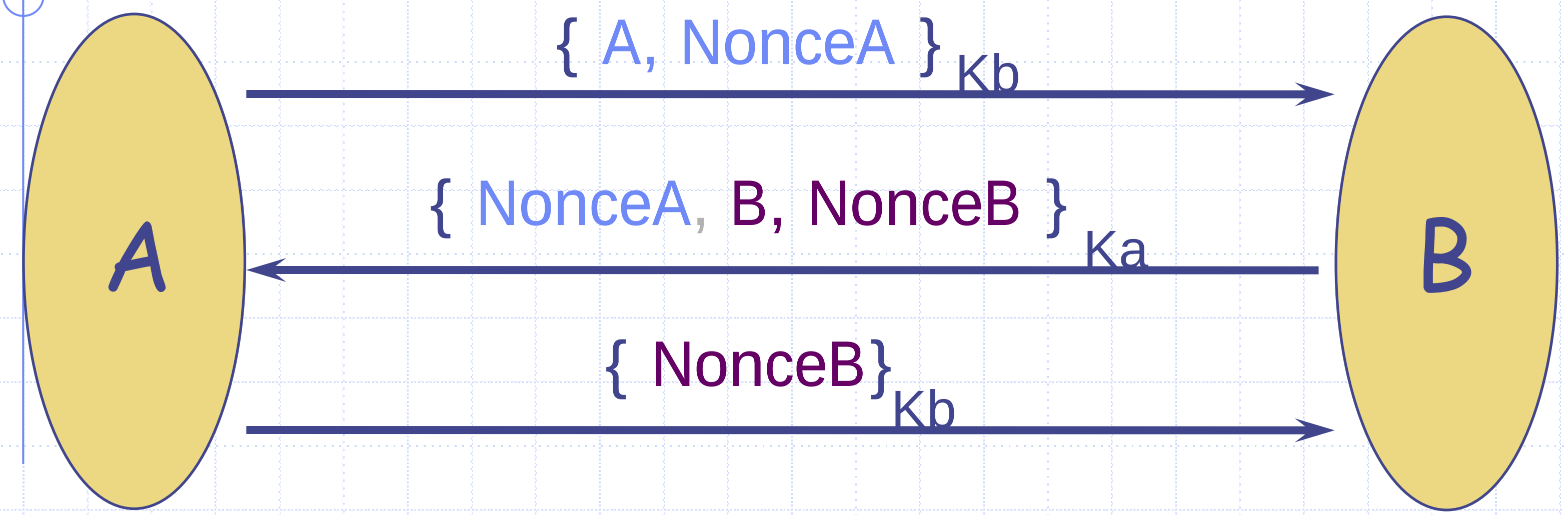
[Lowe]



Evil agent E tricks honest A into revealing private key Nb from B.

Evil E can then fool B.

Needham-Schroeder Lowe



Authentication?

Secrecy?

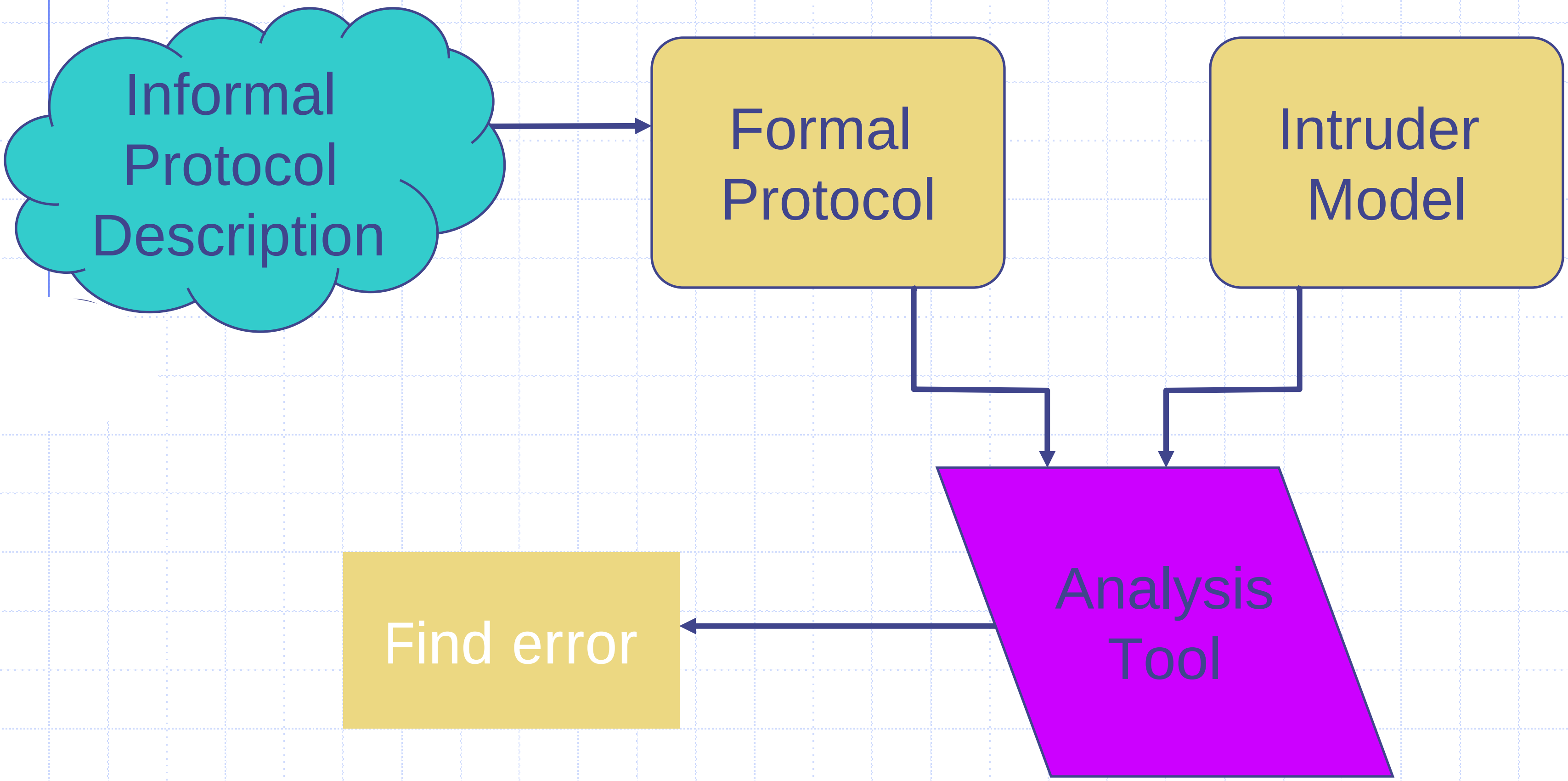
Replay attack

Forward secrecy?

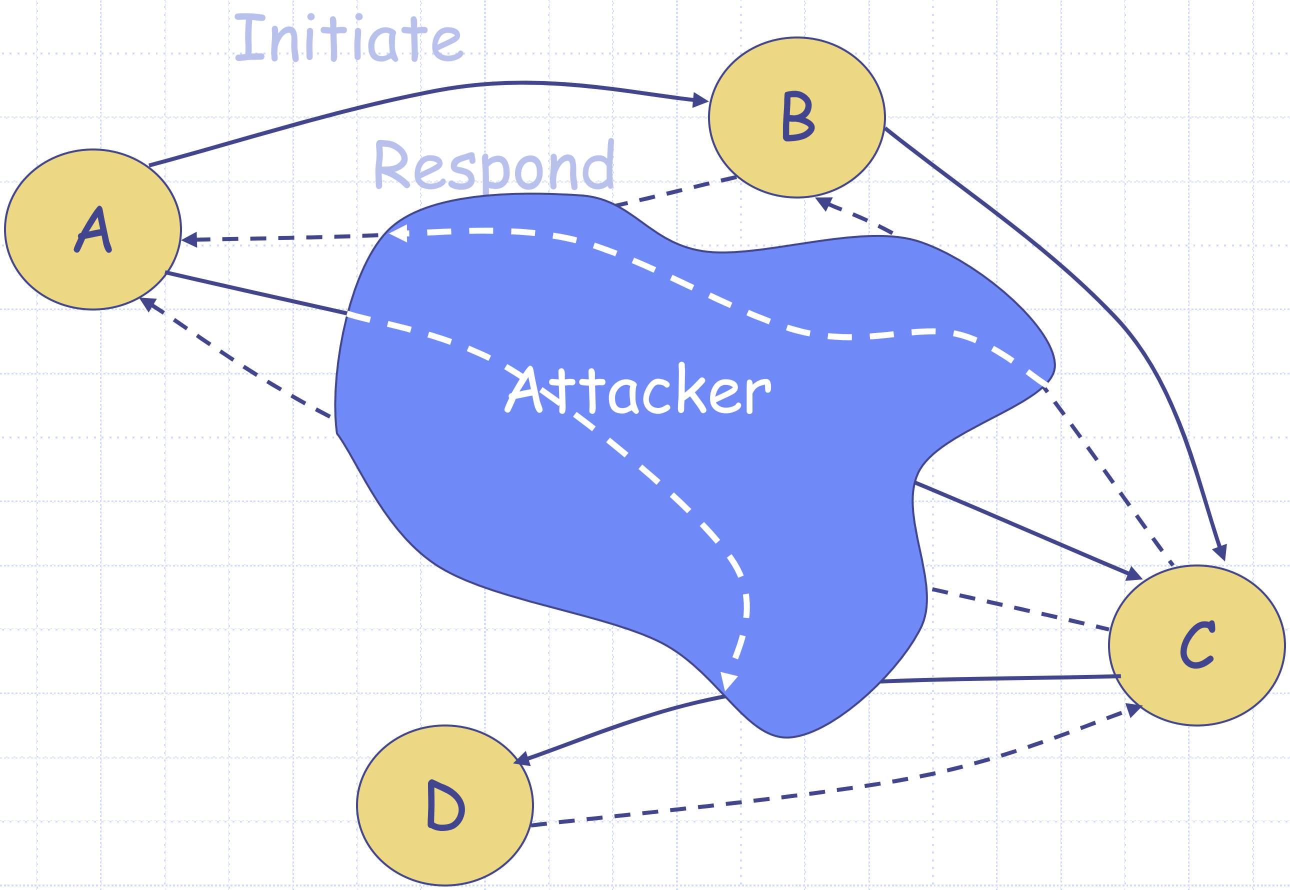
Denial of service?

Identity protection?

Explicit Intruder Method

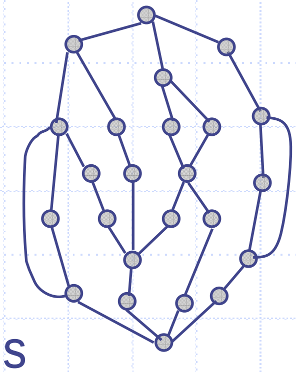


Run of protocol



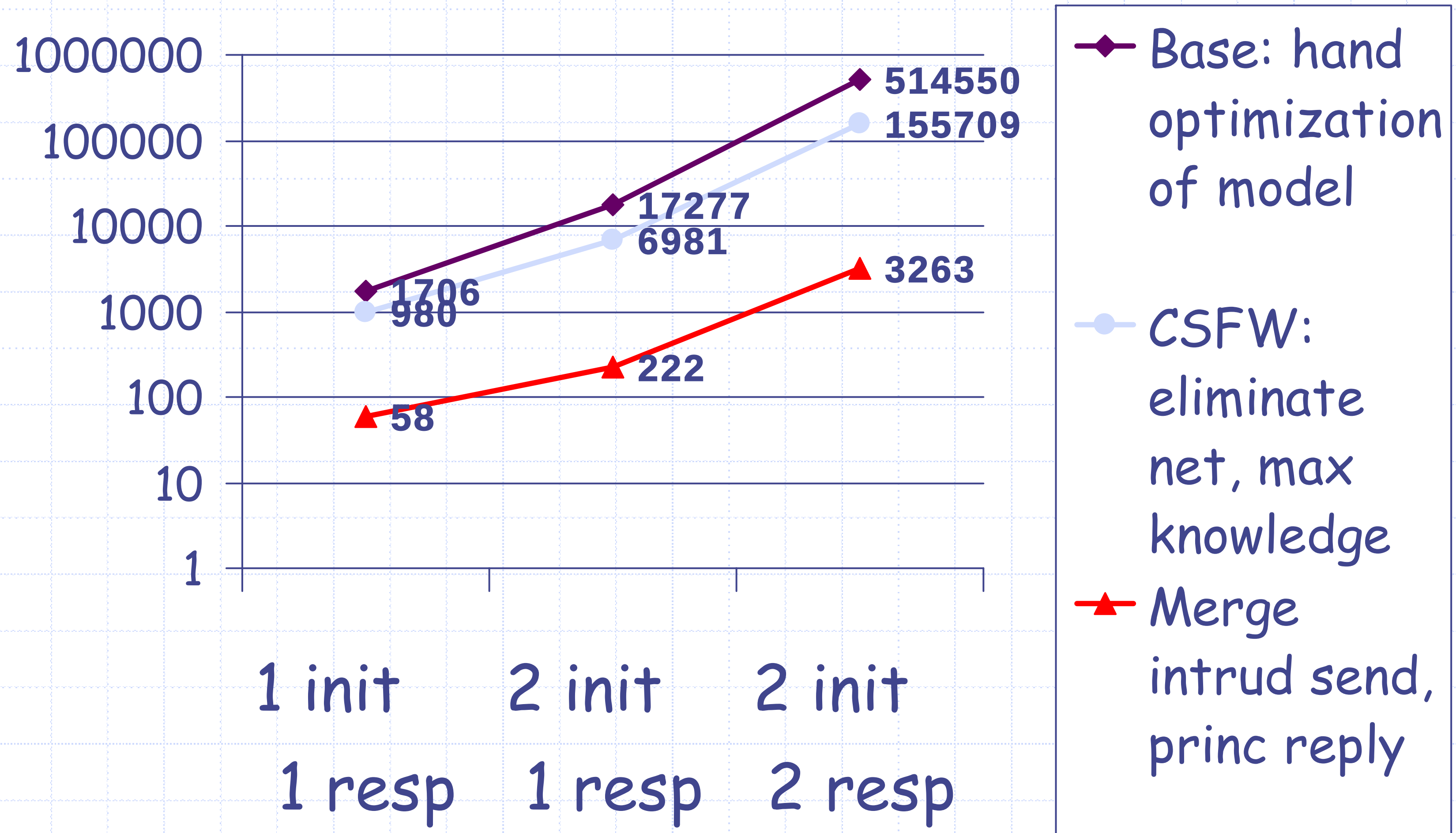
Correct if no security violation in any run

Automated Finite-State Analysis



- Define finite-state system
  - Bound on number of steps
  - Finite number of participants
  - Nondeterministic adversary with finite options
- Pose correctness condition
  - Can be simple: authentication and secrecy
  - Can be complex: contract signing
- Exhaustive search using “verification” tool
  - Error in finite approximation $\Rightarrow$ Error in protocol
  - No error in finite approximation $\Rightarrow$???

State Reduction on N-S Protocol



CS259 Term Projects - 2006

Security Analysis of OTRv2	*Formalization of HIPAA*	Security analysis of SIP
Onion Routing	*Analysis of ZRTP*	MOBIKE - IKEv2 Mobility and Multihoming Protocol
802.16e Multicast-Broadcast Key Distribution Protocols	*Short-Password Key Exchange Protocol*	*Analysis of the IEEE 802.16e 3-way handshake*
Analysis of Octopus and Related Protocols		

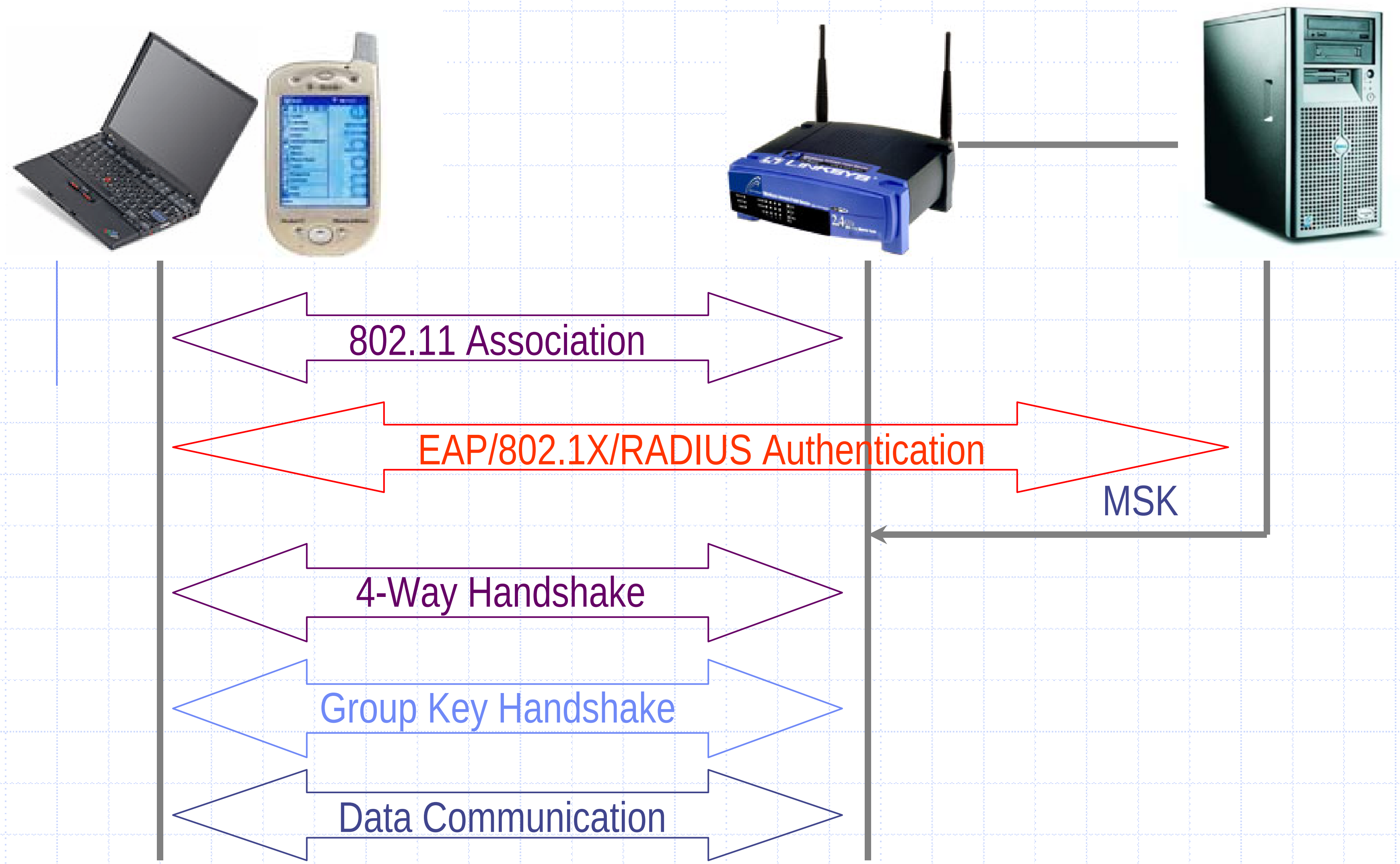
http://www.stanford.edu/class/cs259/

CS259 Term Projects - 2004

iKP protocol family	*Electronic voting*	*XML Security*
IEEE 802.11i wireless handshake protocol	*Onion Routing*	*Electronic Voting*
Secure Ad-Hoc Distance Vector Routing	*An Anonymous Fair Exchange E-commerce Protocol*	*Key Infrastructure*
Secure Internet Live Conferencing	*Windows file-sharing protocols*	

http://www.stanford.edu/class/cs259/WWW04/

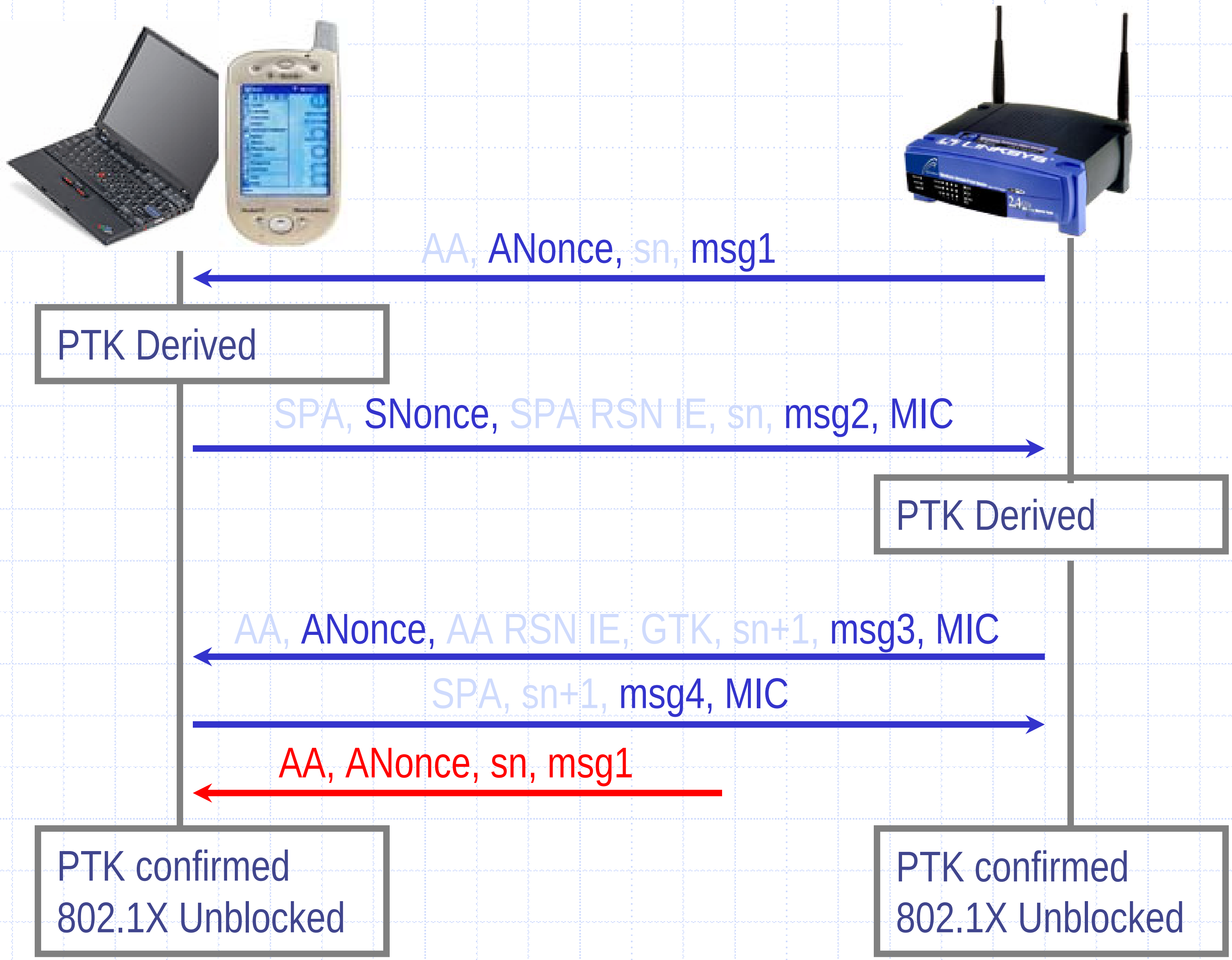
802.11i Protocol



Wireless Threats

- Passive Eavesdropping/Traffic Analysis
  - Easy, most wireless NICs have promiscuous mode
- Message Injection/Active Eavesdropping
  - Easy, some techniques to gen. any packet with common NIC
- Message Deletion and Interception
  - Possible, interfere packet reception with directional antennas
- Masquerading and Malicious AP
  - Easy, MAC address forgeable and s/w available (HostAP)
- Session Hijacking
- Man-in-the-Middle
- Denial-of-Service: cost related evaluation

4-Way Handshake Blocking



Countermeasures

- Random-Drop Queue
  - Randomly drop a stored entry if the queue is full
  - Not so effective
- Authenticate Message 1
  - Use the share PMK; must modify the packet format
- Reuse supplicant nonce
  - Reuse SNonce, derive correct PTK from Message 3
  - Performance degradation, more computation in supplicant
- Combined solution
  - Supplicant reuses SNonce
  - Store one entry of ANonce and PTK for the first Message 1
  - If nonce in Message 3 matches the entry, use PTK directly
  - Eliminate memory DoS, only minor change to algorithm
  - Adopted by TGi

Summary of larger study

ATTACK	SOLUTIONS
security rollback	supplicant *manually* choose security; authenticator restrict pre-RSNA to only insensitive data.
reflection attack	each participant plays the role of either authenti-cator or supplicant; if both, use different PMKs.
attack on Michael countermeasures	cease connections for a specific time instead of re-key and deauthentication; update TSC before MIC and after FCS, ICV are validated.
RSN IE poisoning	Authenticate Beacon and Probe Response frame; Confirm RSN IE in an earlier stage; Relax the condition of RSN IE confirmation.
4-way handshake blocking	adopt random-drop queue, not so effective; authenticate Message 1, packet format modified; re-use supplicant nonce, eliminate memory DoS.

Model checking vs proof

- Finite-state analysis

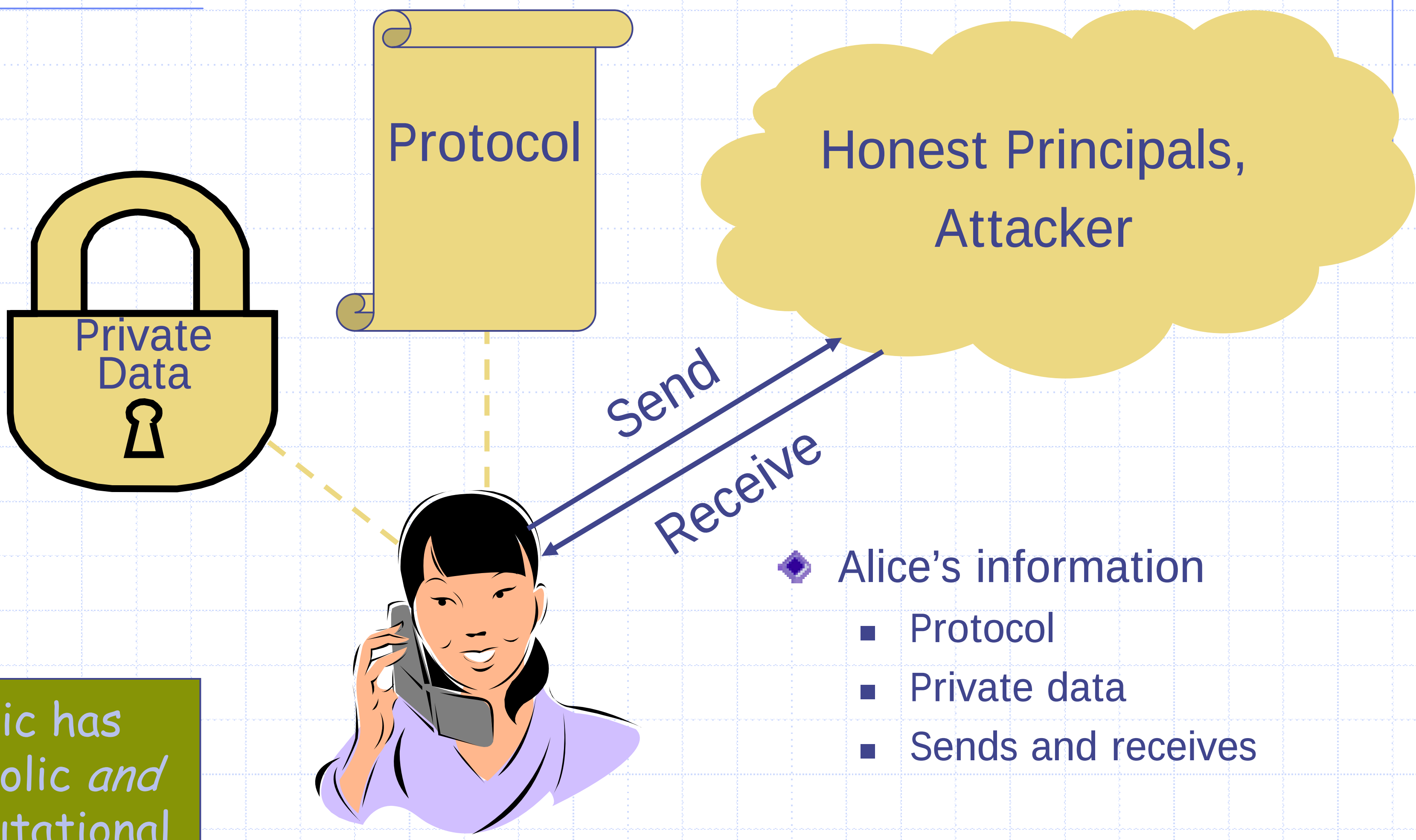
  Attacks on model $\Rightarrow$ Attack on protocol

- Formal proof

  Proof in model $\Rightarrow$ No attack using only these attacker capabilities

Finite state analysis assumes small number of principals, formal proofs do not need these assumptions

Protocol composition logic



- Alice's information
  - Protocol
  - Private data
  - Sends and receives

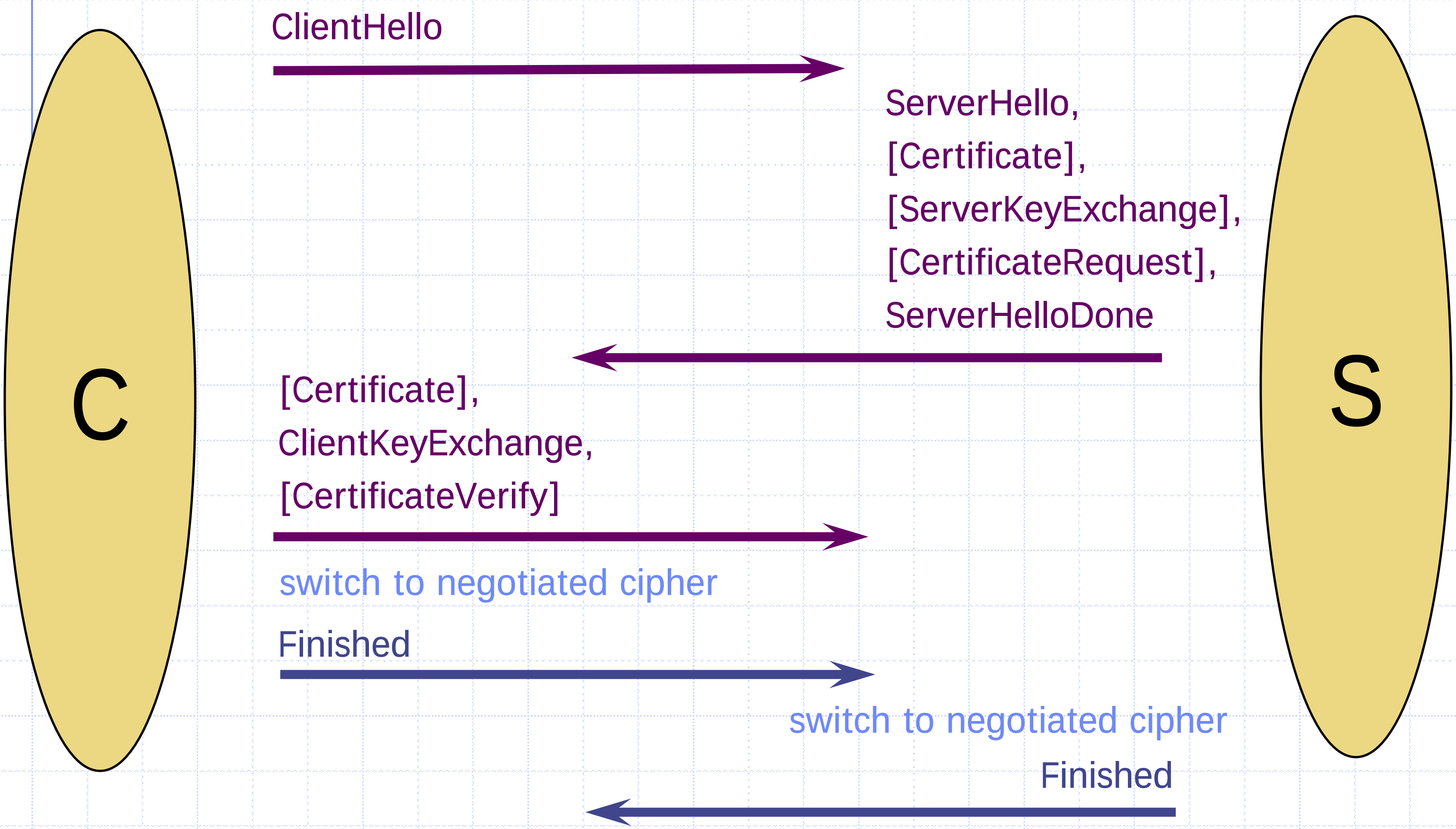
Logic has symbolic *and* computational semantics

802.11i correctness proof in PCL

- EAP-TLS
  - Between Supplicant and Authentication Server
  - Authorizes supplicant and establishes access key (PMK)
- 4-Way Handshake
  - Between Access Point and Supplicant
  - Checks authorization, establish key (PTK) for data transfer
- Group Key Protocol
  - AP distributes group key (GTK) using KEK to supplicants
- AES based data protection using established keys

Formal proof covers subprotocols 1, 2, 3 alone and in various combinations

SSL/TLS


C
S
ClientHello
ServerHello,
[Certificate],
[ServerKeyExchange],
[CertificateRequest],
ServerHelloDone
[Certificate],
ClientKeyExchange,
[CertificateVerify]
switch to negotiated cipher
Finished
switch to negotiated cipher
Finished

Theorems: Agreement and Secrecy

$$\text{Honest}(\hat{X}) \wedge \text{Honest}(\hat{Y}) \wedge \text{Honest}(\hat{CA}) \wedge \hat{X} \neq \hat{Y}$$
$$[\textbf{Client}]_X$$
$$\exists Y.(\text{Send}(X, \hat{X}, \hat{Y}, m1)$$
$$< \text{Receive}(Y, \hat{X}, \hat{Y}, m1)$$
$$< \text{Send}(Y, \hat{Y}, \hat{X}, m2)$$
$$< \text{Receive}(X, \hat{Y}, \hat{X}, m2)$$
$$< \text{Send}(X, \hat{X}, \hat{Y}, m3)$$
$$< \text{Receive}(Y, \hat{X}, \hat{Y}, m3)$$
$$< \text{Send}(Y, \hat{Y}, \hat{X}, m4)$$
$$< \text{Receive}(X, \hat{Y}, \hat{X}, m4))$$

Client is guaranteed:

- there exists a session of the intended server
- this server session agrees on the values of all messages
- all actions viewed in same order by client and server
- there exists exactly one such server session

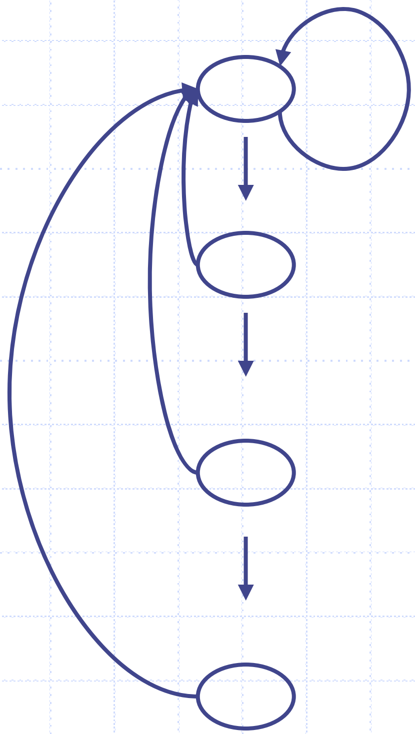
$$\text{Honest}(\hat{Y})[\textbf{Client}]_X$$
$$\text{Has}(\hat{Z}, secret)$$
$$\wedge \hat{X} \neq \hat{Z} \supset \hat{Z} = \hat{Y}$$

Similar specification for server

Composition

- All necessary invariants are satisfied by basic blocks of all the sub-protocols
- The postconditions of TLS imply the preconditions of the 4-Way handshake
- The postconditions of 4-Way handshake imply the preconditions of the Group Key protocol

Complex Control Flows



Simple Flow

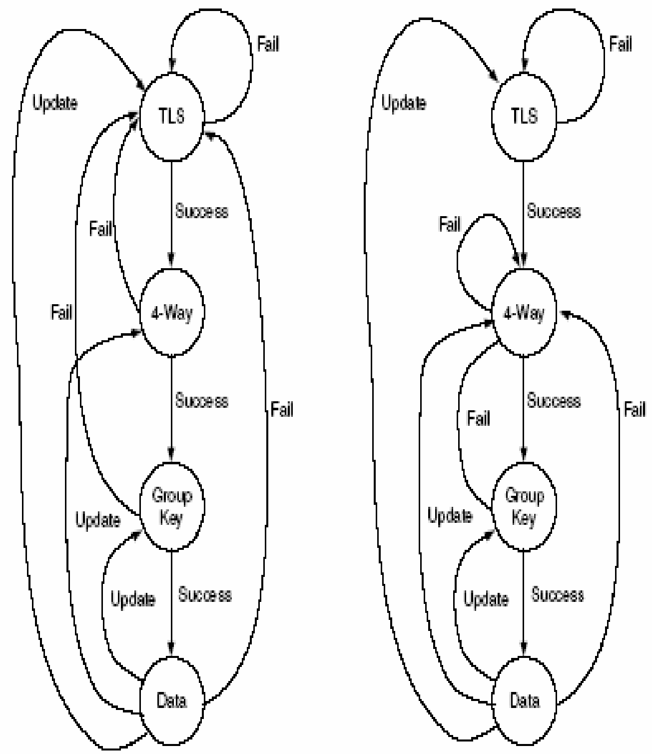


Figure 1: The Control Flow of 802.11i RSNA Establishment Procedure

Complex Flow

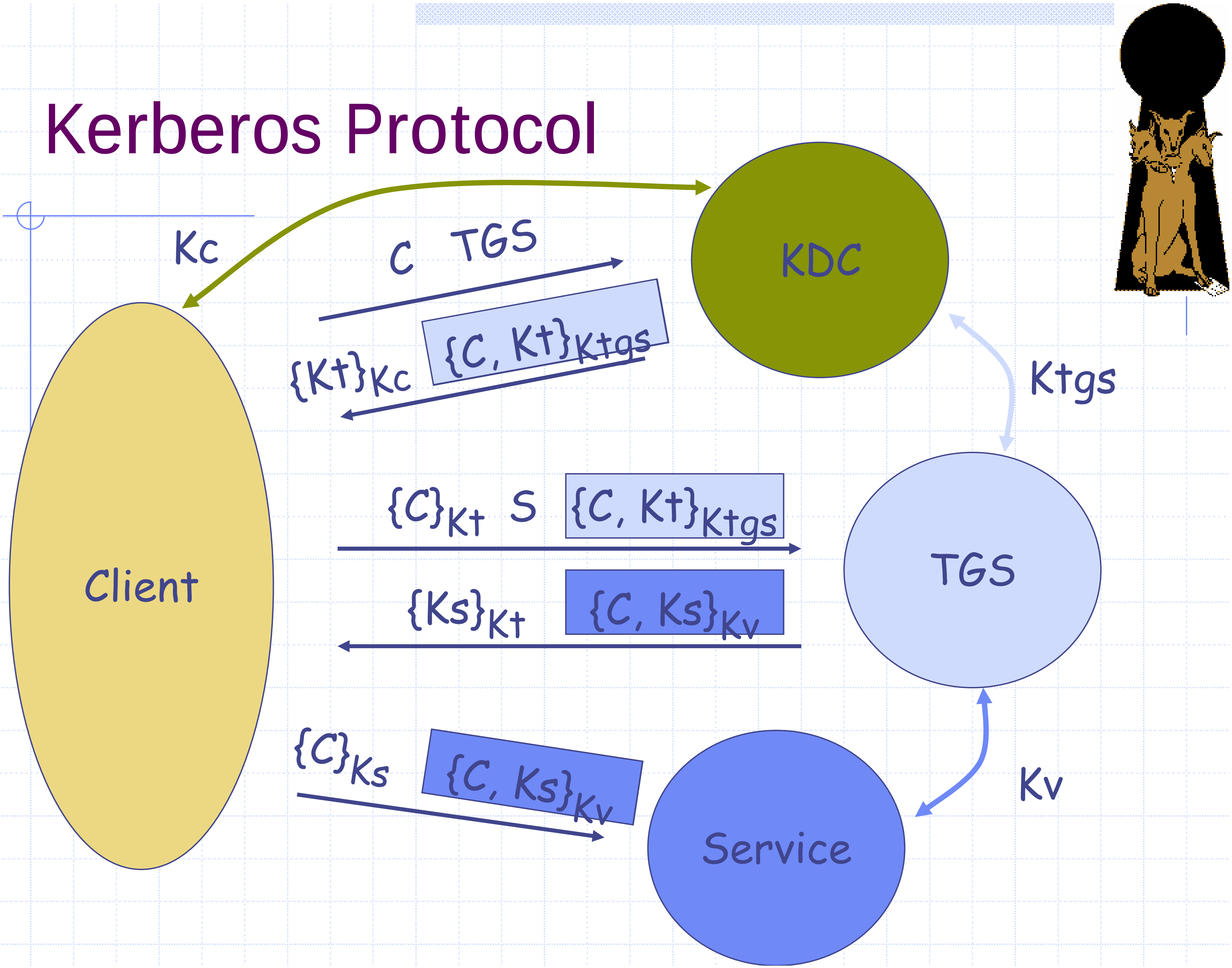
Study results

- 802.11i provides
  - Satisfactory data confidentiality & integrity with CCMP
  - Satisfactory mutual authentication & key management
- Some implementation mistakes
  - Security Level Rollback Attack in TSN
  - Reflection Attack on the 4-Way Handshake
- Availability is a problem
  - Simple policies can make 802.11i robust to some known DoS
  - Possible attack on Michael Countermeasures in TKIP
  - RSN IE Poisoning/Spoofing
  - 4-Way Handshake Blocking
  - Inefficient failure recovery scheme
- Improved 802.11i

Some other case studies

- Wireless networking
  - 802.11i – wireless access point auth
  - 802.16e – metropolitan area networking
  - Simple config – access point configuration
- SSL
  - Found version rollback attack in resumption protocol
- Kerberos
  - PKINIT – public-key method for cross-domain authentication
- IPSec
  - IKEv1, JFK, IKEv2 – improved key management Kerberos
- Mobility
  - Mobile IPv6 – update IP addr to avoid triangle rte
- VoIP
  - SIP – issues with call referral, currently under study
- OTRv2
  - Student project in CS259 this winter

Kerberos Protocol



Microsoft TechNet

Microsoft Security Bulletin MS05-042

Vulnerabilities in Kerberos Could Allow Denial of Service, Information Disclosure, and Spoofing (899587)

Published: August 9, 2005

Affected Software:

- Microsoft Windows 2000 Service Pack 4
- Microsoft Windows XP Service Pack 1 and
  Microsoft Windows XP Service Pack 2
- Microsoft Windows XP Professional x64 Edition
- Microsoft Windows Server 2003 and
  Microsoft Windows Server 2003 Service Pack 1
- Microsoft Windows Server 2003 for Itanium-based Systems and
  Microsoft Windows Server 2003 with SP1 for Itanium-based Systems
- Microsoft Windows Server 2003 x64 Edition

Kerberos Project

I. Cervesato, A. D. Jaggard, A. Scedrov, J.-K. Tsay, and C. Walstad

- Formal analysis of Kerberos 5
  - Several steps
    - Detailed core protocol
    - Cross-realm authentication
    - Public-key extensions to Kerberos
- Attack on PKINIT
  - Breaks association of client request and the response
  - Prevents full authentication and confidentiality
- Formal verification of fixes preventing attack
  - Close, ongoing interactions with IETF WG

Public-Key Kerberos

- Extend basic Kerberos 5 to use PKI
  - Change first round to avoid long-term shared keys
  - Originally motivated by security
    - If KDC is compromised, don't need to regenerate shared keys
    - Avoid use of password-derived keys
  - Current emphasis on administrative convenience
    - Avoid the need to register in advance of using Kerberized services
- This extension is called PKINIT
  - Current version is PKINIT-29
  - Attack found in -25; fixed in -27
  - Included in Windows and Linux (called Heimdal)
  - Implementation developed by CableLabs (for cable boxes)

The Attack

At time t_C, client C requests a ticket for ticket server T (using nonces n_1 and n_2):

C $\xrightarrow{Cert_C,\ [t_C,\ n_2]_{skC},\ C,\ T,\ n_1}$ I

The attacker I intercepts this, puts her name/signature in place of C's:

I $\xrightarrow{Cert_I,\ [t_C,\ n_2]_{skI},\ I,\ T,\ n_1}$ K

Kerberos server K replies with credentials for I, including: fresh keys k and AK, a ticket-granting ticket TGT, and K's signature over k,n_2:

(Ignore most of enc-part)

I $\xleftarrow{\{[k,\ n_2]_{skK}\}_{pkI},\ I,\ TGT,\ \{AK,\ \ldots\}_k}$ K

I decrypts, re-encrypts with C's public key, and replaces her name with C's:

C $\xleftarrow{\{[k,\ n_2]_{skK}\}_{pkC},\ C,\ TGT,\ \{AK,\ \ldots\}_k}$ I

- **I knows fresh keys k and AK**
- C receives K's signature over k,n_2 and assumes k, AK, etc., were generated for C (not I)

- Principal P has secret key skP, public key pkP
- $\{msg\}_{key}$ is encryption of msg with key
- $[msg]_{key}$ is signature over msg with key

Fix Adopted in pk-init-27

- The KDC signs k, cksum (place of k, n_2)
  - k is replyKey
  - cksum is checksum over AS-REQ
  - Easier to implement than signing C, k, n_2
- Formal proof: this guarantees authentication
  - Assume checksum is preimage resistant
  - Assume KDC's signature keys are secret

Published proof uses simplified symbolic model

Cryptographically sound proofs now exist

Recent and ongoing protocol efforts

- Wireless networking authentication
  - 802.11i – improved auth for access point
  - 802.16e – metropolitan area networks
  - Simple config – setting up access point
  - Bluetooth simple pairing protocols
- Mobility
  - Mobile IPv6 – update IP addr to avoid triangle routing
- VoIP
  - SIP – call referral feature, other issues
- Kerberos
  - PKINIT – public-key method for cross-domain authentication
  - Full cryptographically sound proof recently developed
- IPSec
  - IKEv1, JFK, IKEv2 – improved key management
- OTRv2
  - student project in CS259 this winter
  - ZPhone ??

Conclusions

- Protocol analysis methods
  - Model checking is fairly easy to apply
  - Ready for industrial use
  - Logical proofs are feasible, can be made easier
- Example: Wireless 802.11i
  - Automated study led to improved standard
  - Deployment recommendations, more flexible error recovery
- Many ongoing efforts
  - Examples: Wireless networking, VoIP, mobility
  - Typical standardization effort takes a couple of years

Achievable goal: systematic methods that can be used by practicing engineers to improve network, system security