Distributed CA-based PKI for Mobile Ad hoc Networks using Elliptic Curve Cryptography

Charikleia Zouridaki[1], Brian L. Mark[1], Kris Gaj[1],
Roshan K. Thomas[2]

[1] George Mason University, Electrical and Computer Engineering Dept., 4400 University Drive, Fairfax, VA, 22030, USA
{czourida, bmark, kgaj}@gmu.edu

[2] McAfee Research, Network Associates,Inc., 1145 Herndon Parkway, Suite 500, Herndon, VA, 20170, USA
RThomas@nai.com

Agenda

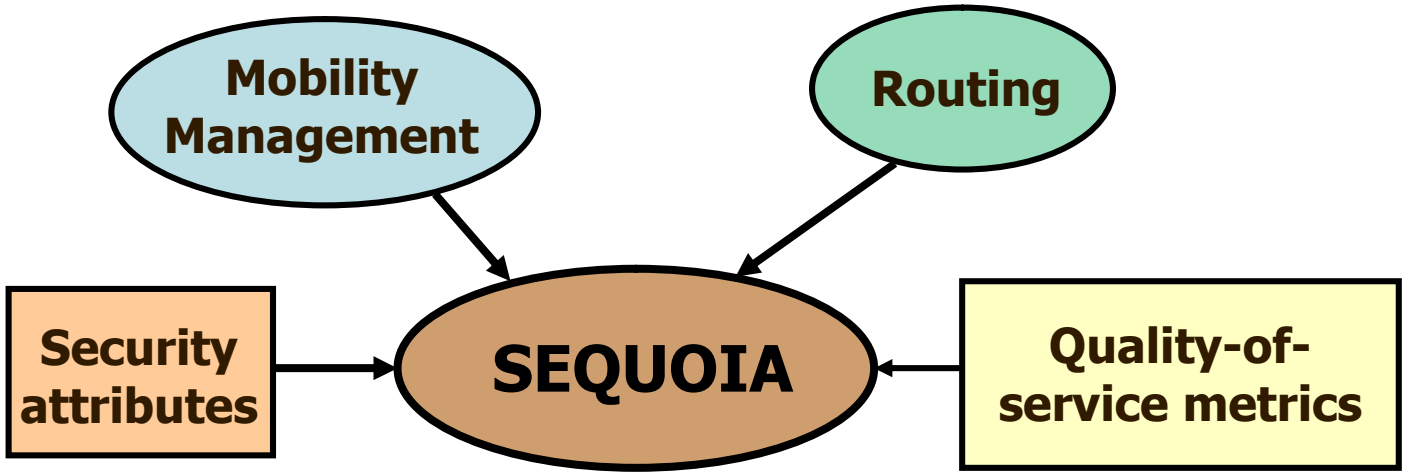
1. Introduction
2. Distributed CA-based PKI Architecture
3. Operational Aspects of the Proposed Architecture
4. Performance Gains using ECC
5. Conclusions

George Mason University

Background: The SEQUOIA Project:

SEcurity-and-Quality-Of-Service-In-Mobile-Adhoc-Networks

(NSF funded, http://sequoia.gmu.edu)



- How do we integrate security and QoS requirements into routing protocols for MANETS?
- How can we meaningfully tradeoff security and QoS?
- How can security and QoS-aware routing be optimized through the use of mobility information?

George Mason University

Agenda

1. Introduction
   - Mobile Ad hoc NETworks (MANETs)
   - Security in MANETs - Problem Statement
   - Related Work
   - Key Elements of our approach

Introduction

- MANETs:
  - IETF definition: a MANET is an autonomous system of mobile routers (and associated hosts) connected by wireless links; the union of which forms an arbitrary graph

- Security in MANETs:
  - Challenging problem due to:
  1. lack of a fixed infrastructure
  2. dynamically changing network topology
  3. limitations of the wireless channel (intermittent and unreliable nature)
  4. limited capabilities of the nodes

Introduction – Problem Statement

- Why a standard PKI is not practical in a MANET:
  - a centralized certification authority (CA) represents a single point of failure in the network
  - MANETs cannot always guarantee online access to a centralized CA
    - due to the often intermittent and unreliable nature of the wireless channel
- PKI architectures designed with wired networks in mind cannot be carried over straightforwardly to MANETs

Introduction - Related Work

- Password-based schemes for key establishment (Jablon etc. 1992-1999)
  - carry out auth. on the basis of a shared secret or passwd established prior to the network deployment
- A security scheme similar to PGP (Hubaux, Buttyan, Capkun, Terminodes project, 2001)
  - C issued by users, establishment of chains of trust
  - suited to MANETs, but probabilistic guarantees, relies on transitive trust relationships
- Distributed CA (Zhou, Haas, 1999)
  - avoids the single point of failure, deterministic guarantees, but raises scalability issues

Introduction – Key Elements of our approach

1. dynamically partitioning the network into smaller clusters of nodes [P. Basu et al., S. Banerjee et al., C.R. Lin et al., etc.]
   - reduces storage requirements of nodes/CA servers, computational/signaling overhead

2. distributed CA with multiple CA servers employing threshold-based cryptography with proactive share recovery (as proposed by Jarecki, 1995)
   - Compromised CA servers > 1/2 → distributed CA compromised

Introduction – Key Elements of our approach

3. distributed geographically the CA servers
   - difficult for adversary to compromise multiple CA servers simultaneously

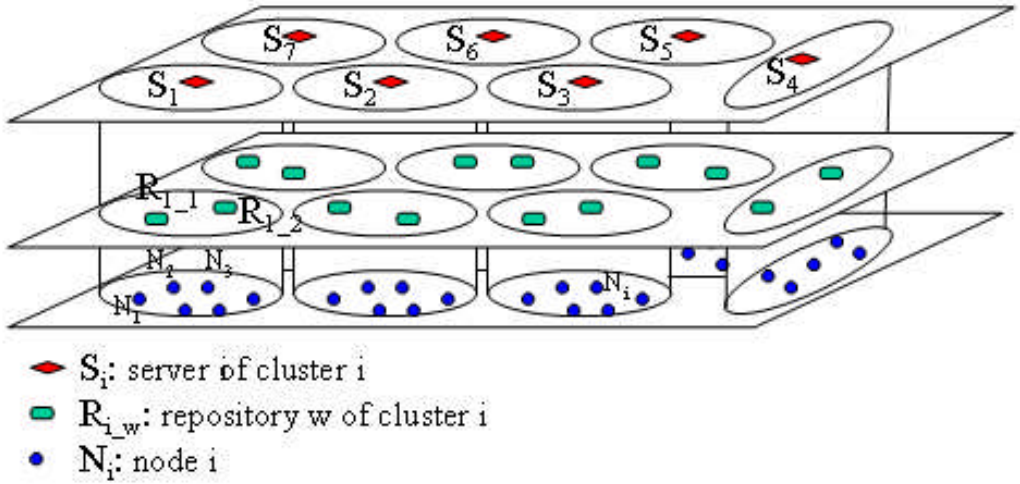
4. replicated key repositories assigned to each cluster
   - Active Repositories are guaranteed → nodal comm. is not interrupted

5. use of elliptic curve cryptography (ECC)
   - reduces the computations of cryptographic operations

Agenda

2. Distributed CA-based PKI Architecture
   - Overview
   - Elliptic Curve-based Distributed CA

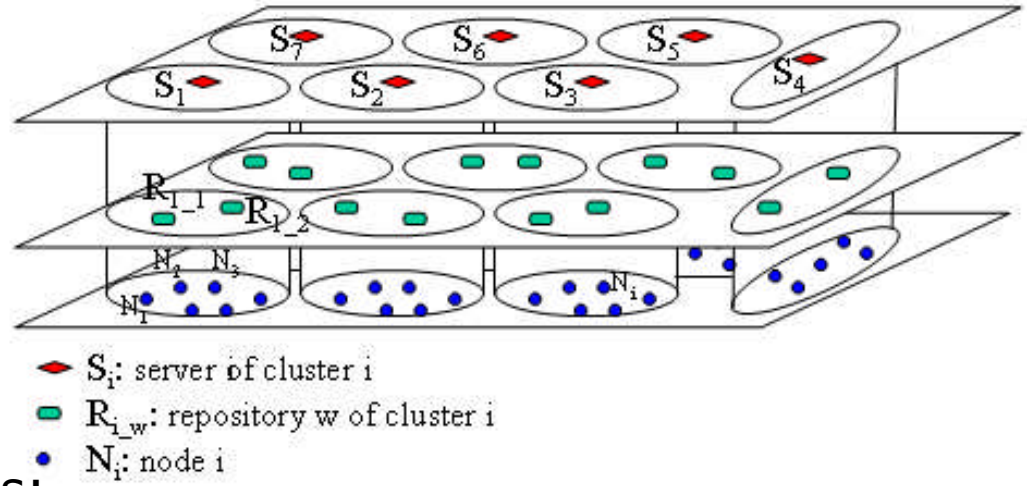
Distributed CA-based PKI Architecture



- Servers:
  - DCA server #= n = 2k+1, where k = max DCA server # that can be compromised in a predefined period of time
    - [n=f(network_size, resilience_degree_required)]
  - Each server participates in issuing and revoking certificates and in signing the CRL
  - The servers are assumed to be physically more secure and computationally more powerful nodes

George Mason University

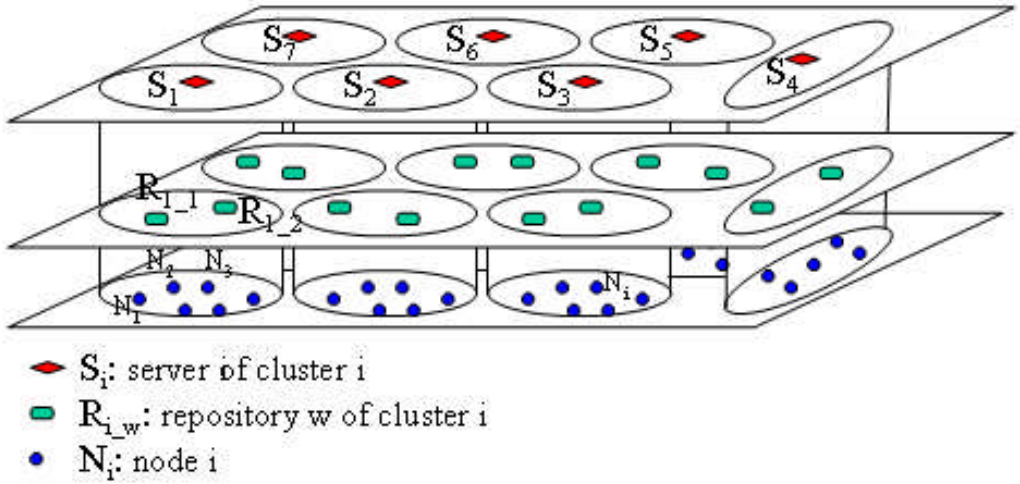
Distributed CA-based PKI Architecture



- Repositories:
  - t nodes are designated as repositories within each cluster
  - The repositories store:
    1. C of the cluster nodes
    2. C of CA servers
    2. CC of network nodes
    4. recent CRL
  - The corruption of the repositories is acceptable, since
    - corrupted C or CC will be detected via signature verification
    - corruption is detected checking the most recent CRL
    - → up-to-date info is considered correct
  - Guaranteed a minimum of t - r active repositories in each cluster

George Mason University

Distributed CA-based PKI Architecture



- Nodes store:
  1. C of cluster nodes, as needed
  2. CC of network nodes
  3. recent CRL
  4. C of DCA, system parameters
  5. Server signature verification key, encryption key

Elliptic Curve-based Distributed CA

- Time divided into:
  - Time Periods
    - servers perform the group signature operation (1)
  - Update Phase
    - share renewal (re-randomizes the original key) (5)
    - private key renewal (2)
    - lost share detection and recovery (to reconstruct the corrupted shares, if any) (3, 4)
    - resolving accusations (when contradictory servers, in 3 or 4 steps) (6, 7)

Agenda

3. Operational Aspects of the Proposed Architecture

- Network Initialization
- ☑ Activating a Node
- Deactivating a Node
- Node Migrations across Clusters
- ☑ Intra-Cluster Communications
- ☑ Inter-Cluster Communications
- Cluster splitting and merging

Operational Aspects of the Proposed Architecture

- Activating a Node
- Node N obtains certificate c_N
  - N contacts RA
  - RA verifies credentials of N and securely contacts k+1 CA servers
  - (k+1) CA servers issue c_N for N and send it to RA
  - RA gives c_N and c_{DCA} to N
- Node N joins cluster i
  - N sends c_N to Ri_w
  - N requests Ci , CC, CRL from Ri_w
  - Ri_w broadcasts c_N
  - Ri store c_N
  - ji optionally store c_N

Operational Aspects of the Proposed Architecture

- Intra-Cluster Communications
  - Encrypted / Authenticated = fast
- Inter-Cluster Communications
  - Authenticated = fast
  - Encrypted: required C has to be requested
- Revoked C will never be requested → reduction of communication overhead

Agenda

4. Performance Gains using ECC

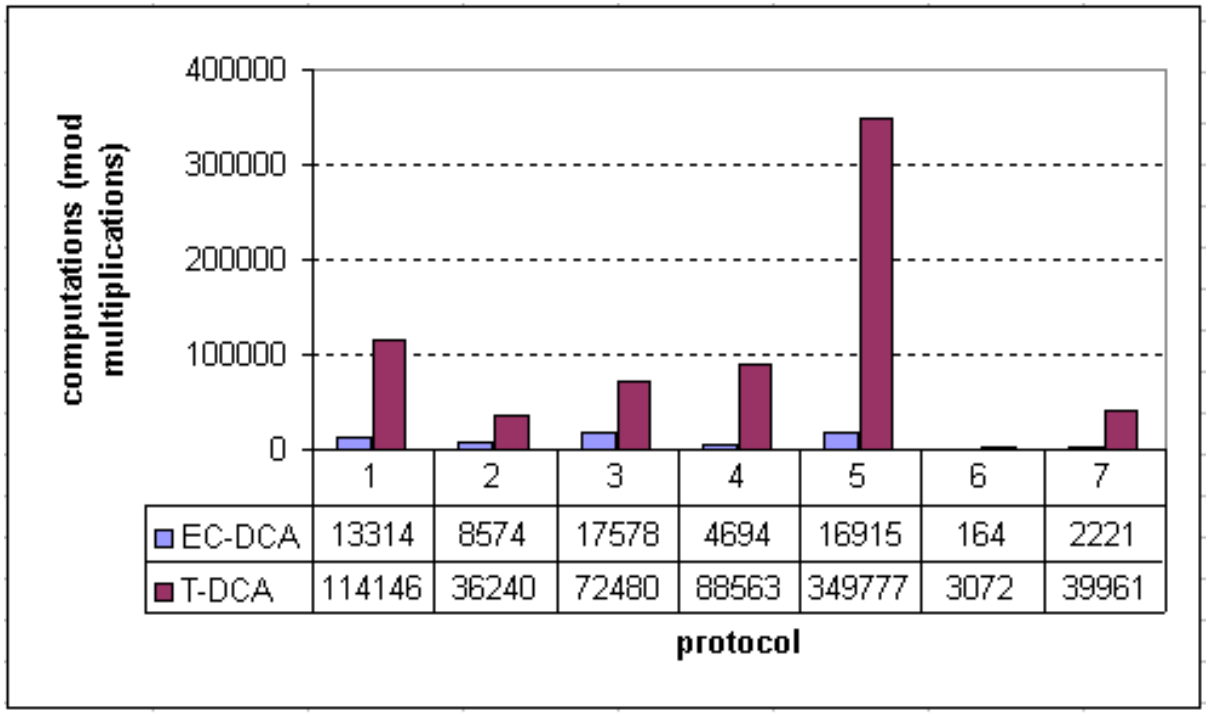
   - Cryptographic Tools
   - Computational and Time Requirements of the DCA
   - Storage Requirements

Performance Gains using ECC

- ECC = appropriate for mobile nodes, AS it requires smaller keys, involves operations on smaller integers
- Cryptographic Tools:
  - EC ElGamal Signatures, signature scheme of DCA
  - ECDCA (EC Digital Signature Algorithm), signatures
  - EC ElGamal Encryption, encryption
- 160 bits key size (= 1024 bit key in traditional public-key crypto)
- Since: 160 bit key provides a sufficient level of security for most applications, while placing a reasonable computational burden on the nodes of a MANET

Performance Gains using ECC

- Computational Requirements of a DCA server (proportional to k^2)
- network of 51 servers (k = 25); 100 nodes/cluster → network of 5100 nodes & 51 servers
- Comparison of the max computations per server for each protocol



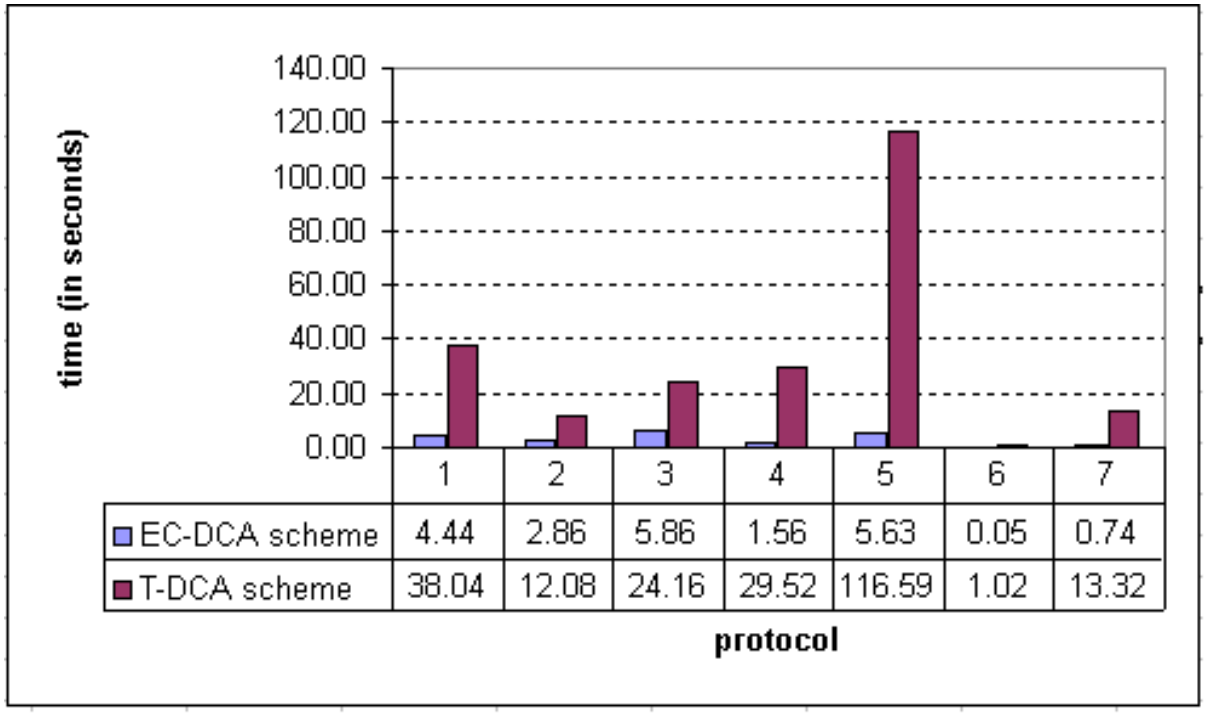
	1	2	3	4	5	6	7
EC-DCA	13314	8574	17578	4694	16915	164	2221
T-DCA	114146	36240	72480	88563	349777	3072	39961

Performance Gains using ECC

- Time Requirements of the DCA

- Smart cards perform up to 3000 modular multiplications/sec with the size of the modulus being 1024 → we can calculate time required for each protocol

- Comparison of the time required for each protocol



	1	2	3	4	5	6	7
EC-DCA scheme	4.44	2.86	5.86	1.56	5.63	0.05	0.74
T-DCA scheme	38.04	12.08	24.16	29.52	116.59	1.02	13.32

Performance Gains using ECC

- Storage Requirements

- key space required for: C, CC, CRL & DCA parameters / server, repository & node
  - Max key space required/node = /repository

- If C=100, CC=20/cluster, with 32 bits sequence # → CRL=4.04KB
  - Per server=29.66KB
  - Per node/repository= 28.6KB

Agenda

5. Conclusions

- Our distributed CA-based PKI architecture addresses:
  1. physical vulnerability of the nodes
     - addressed by employing distributed-CA based PKI, using threshold cryptography with proactive recovery
  2. insecurity of the wireless links
     - dealt with the use of keys → information exchanged is authenticated/encrypted
  3. storage constraints
     - addressed with use of ECC (the key size is reduced) & clusters (the number of keys stored is reduced)
  4. energy constraints
     - addressed with use of ECC-based cryptosystem & clustering (to reduce communication overhead, as Cs/ CA servers available within a small hop#)

Conclusions

- The use of clustering allows the proposed PKI scheme to scale to large networks
- The proposed architecture could be implemented using current smartcard technology

  [Rankl and Effing, Smart Card Handbook, 2000]

Thank you! ☺

Questions?

George Mason University

Analysis of computational gain

- The performance of one modular multiplication in software is given by formula: $T_{MULT} = c*k^2$,
- where c is a constant and k the size of the modulus

- Therefore: $$\frac{T_{MULT}(1024)}{T_{MULT}(160)} = \frac{c*k^2}{c*l^2} = \frac{k^2}{l^2} = \frac{1024^2}{160^2} = 40.96$$

- Modular multiplications with modulus of the size of 160 bits can be normalized to modular multiplications for a 1024 bit modulus, to make the comparison simple

Registration Authority (RA)

- We assume a RA that
  1. is part of a wired network
  2. can communicate with the DCA servers securely
  3. does not know the private key of the DCA

  The RA verifies the credentials of a node & if satisfied, contacts at least k+1 servers and requests issuance of C

- This model reflects the practical procedures and work scenarios present in many environments that use wireless networking
  - E.g.: troops going out to the battlefield (required to report & register at the RA)