

A Cluster-Based Security Architecture for Ad Hoc Network

Represented by: Thu Trinh, Chau
Ngo

Supervised by : Dr. Ratan Guha

OUTLINE

- ◆ Authors
- ◆ Why cluster-based?
- ◆ Why difficult to implement security in Ad Hoc networks?
- ◆ Clustering
- ◆ Conceptual Building Blocks
- ◆ Details and Protocol
- ◆ Evaluation
- ◆ Conclusion

Authors

- ◆ M. Bechler, H.-J. Hof, D. Kraft, F. Pählke, L. Wolf
- ◆ German researchers.

Why cluster-based?

- ◆ Common authentication schemes are not applicable in Ad hoc network since public key infrastructure are hard to deploy.
- ◆ Central authority reach ability not guaranteed

Why difficult to implement security in Ad Hoc Networks?

- ◆ High dynamics of topology: mobility, joining/leaving devices.
- ◆ Limited resources of end systems
- ◆ Bandwidth restricted
- ◆ Possibly asymmetric communication links

Clustering

- ◆ **Type of nodes:**
 - Cluster heads
 - Gateways
 - Nodes
- ◆ **Responsibilities:**
 - Cluster head: sends beacons containing list of nodes and gateways and their status
 - . execute administrative functions
 - . hold shares of network key used for certification.
 - . establish and organize cluster
 - . broadcast network public key and their own public key within their clusters

Clustering

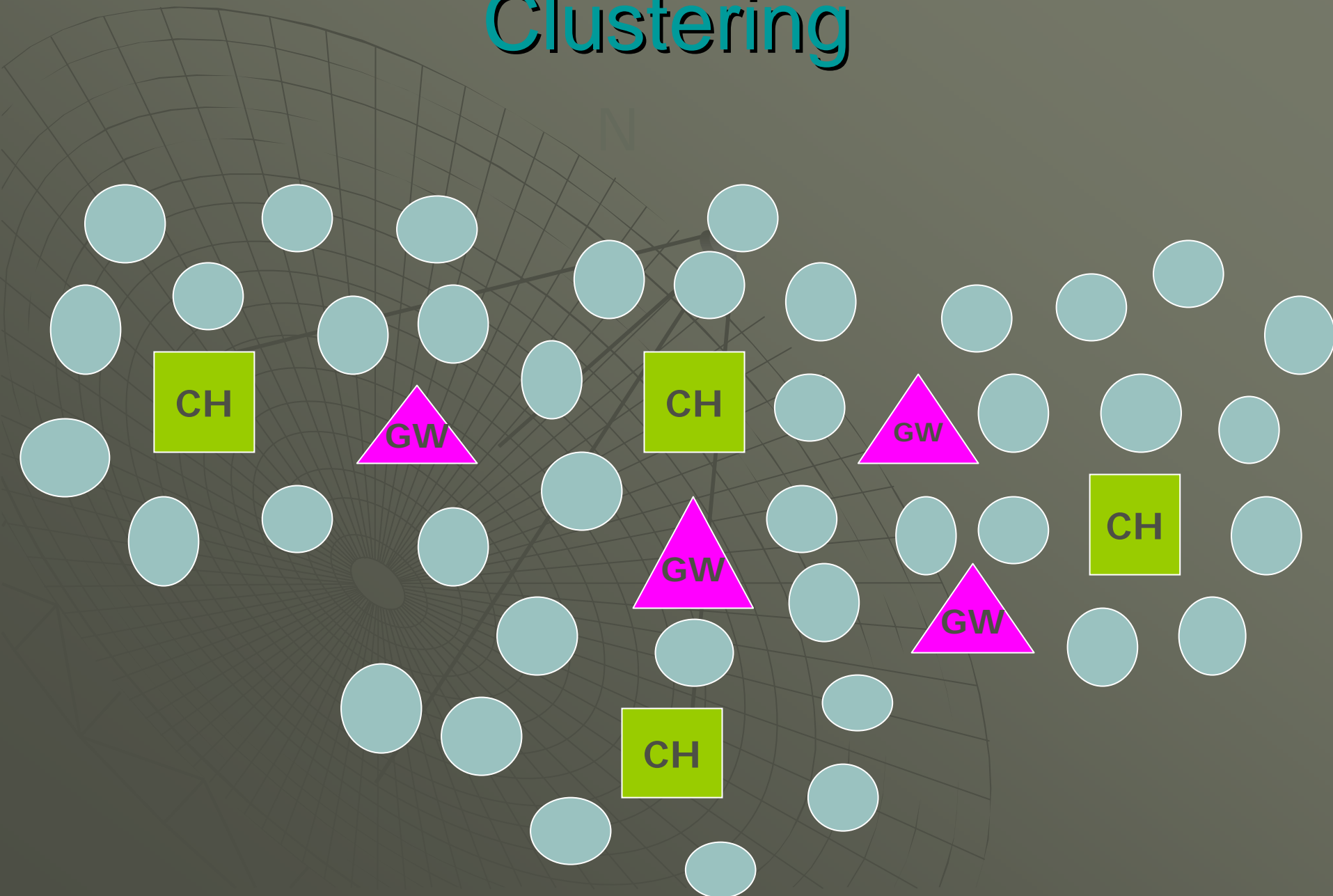
- Gateway:

- . manages communications with adjacent clusters
- . send beacons about adjacent clusters

- Node: full member can

- . warrant for new nodes
- . manage certain resources and /or services

Clustering



Clustering

REGULATIONS:

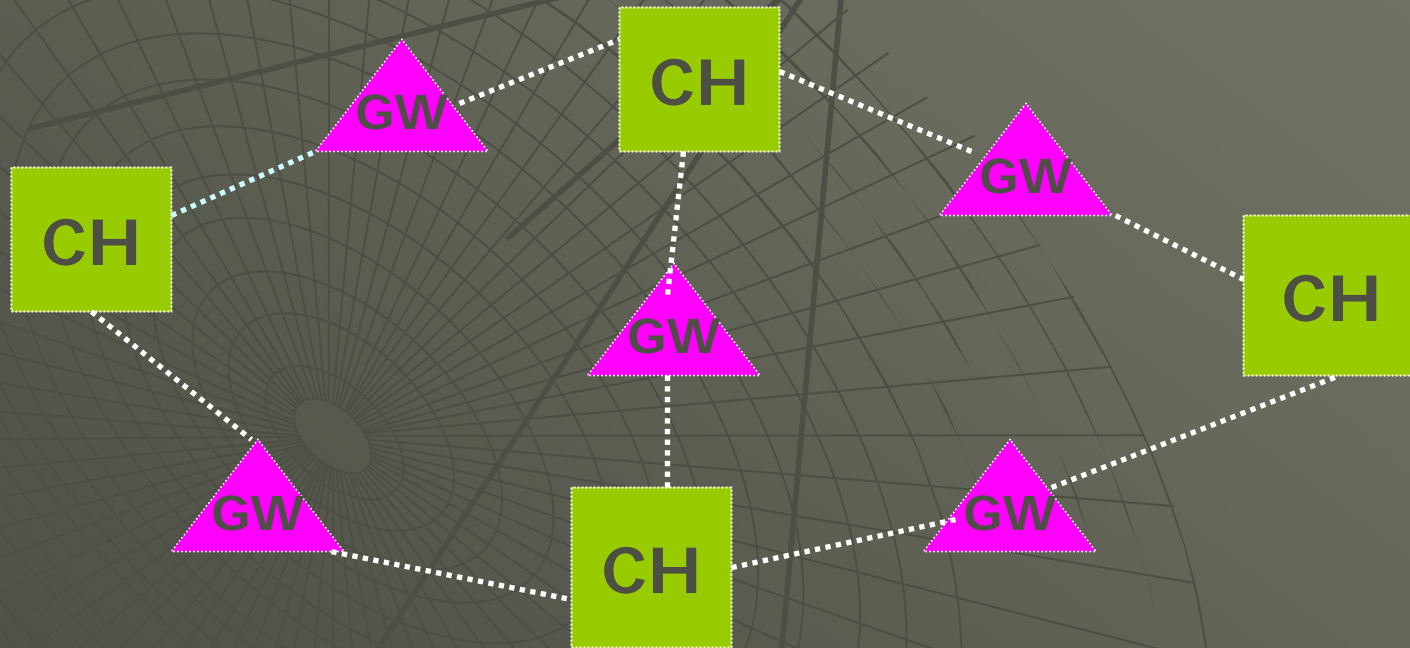
- ◆ Intra-cluster: routing within cluster
- ◆ Inter-cluster: routing between different clusters; only via gateways.
- ◆ Nodes cannot find cluster forms some themselves
- ◆ Existing clusters are merged and split on demand
- ◆ **Inter-cluster communication:** use public key
- ◆ **Intra-cluster communication:** use symmetric encryption

Conceptual Building Blocks

- ◆ **NETWORK-WIDE CERTIFICATION INFRASTRUCTURE:**
 - Every node holds a self-generated key pair.
 - Distributed certification authority (CA):
 - . enhances availability (threshold (n,k))
 - . security infrastructure becomes more resistant against intruder.
 - Cluster heads from logical CH-network: every CH hold a fragment of the whole private key of CA (= network key).
 - Overlap networks: allowed; clustering is independent from routing; CHs only hold fragments of 1 network key.

Conceptual Building Blocks

Logical CH-network



Conceptual Building Blocks

- ◆ **INTRA-CLUSTER SECURITY:**
 - Use symmetric key.
 - Protect traffic on links, eavesdroppers
 - Hide info

Conceptual Building Blocks

- ◆ **NODE STATUS AND AUTHORIZATION:**

- **Status:**

- . Guest: no access right
 - . Full member: after CH-network signed its public key.

- **Authorization:**

- . initial authentication: obtain warranty cert from neighbor nodes. The more warrant cert than minimum collected -> more access rights.
 - . after becoming full member, node can acquire additional access rights by having authorization cert.
 - . Authorization cert can be issued by any network node who are managing a service or resource e.g. printer, Internet access.

Details and Protocols

- ◆ **KEY DISTRIBUTION AND KEY REFRESHMENT:**

- **Distribution:** number of key share adapt to number of cluster heads

- Use Proactive secret sharing

- **Refreshment:** after certain period of time

Details and Protocols

- ◆ **LOG-ON PROCEDURE:** new node A
 - **find a cluster**
 - ***if cannot find any cluster:*** forms its own cluster and acts as cluster head; generate secret cluster key and starts transmitting CH beacon. ***In this case, new node is not authenticated. Need some work on algorithm!***
 - ***if receive CH beacons:*** follow the steps
 1. A sends log-on request to CH. A and CH negotiate the number of warranty cert (WC) required; how cluster key can be used. A is a guest
 2. A requests S for S's WC.
 3. S verifies A's identity: by visual, physical, contact or voice communication, or external CA

4. S sends A its warranty cert. from full member S
$$WC(A) := \text{node}(A), \text{PubKey}(A), \text{Validity}(t),$$
$$\text{Fct}(\text{"S warrants for A"}), \text{Sign}(S)$$

S also sends A its Warranty Authorization Certificate

$$WAC(s) := \text{node}(S), \text{PubKey}(S), \text{Fct}(\text{"S may warrant"}), \text{Sign}(\text{CH-network})$$
5. A sends both to CH
6. CH sends both certificates to k CHs make sure issuer (S) is authorized to vouch for a guest
7. If both certs are valid, each CH send its shares of an identity cert. back to A.

Details and Protocols

8. If A collects enough cert shares, it can complete its identity certificate.

$\text{IdCert}(A) := \text{Node}(A), \text{PubKey}(A), \text{Validity}(t),$
 $\text{Sign}(\text{CH-network})$

Note: A needs to know at least k CHs in a (k, n) threshold scheme. A can find info about other CHs through GWs or its own CH.

9. A becomes full member. CH sends cluster symmetric key to A.

Note: how easy or difficult for a node to find warrants: depends on the distribution of warrants. Assume:

- there is no limit for this process.
- a full member is granted warranting privilege after some time

Log-on Procedure

T

A

S

CH

GW

CH

GW

GW

CH

Details and Protocols

- ◆ **INTERACTION WITH ROUTING:**
 - Set flag in routing header to specify if either only full members or all nodes are allowed to forward a packet
 - Each CH defines the cluster's security guidelines

Details and Protocols

◆ GATEWAYS:

- each node that gets in contact with a foreign cluster can act as a gateway. It needs to obtain gateway authorization certificate signed by the CH network first

$GwAutCert := Node(N), PubKey(N),$
 $Fct("Gateway"), Sign(CH-network)$

- new GW notifies its CH and discovered CH, other nodes in discover cluster.

Details and Protocols

- ◆ **DELEGATION OF CLUSTER HEADS:**
 - Find trusted node
 - Securely migrates its states to successor, transfer its share of private network key.
 - Send message to nodes in clusters about new CH identity
 - Nodes that do not receive the message will consider the new CH beacons as foreign
 - Notifies members of CH network

Details and Protocols

◆ MERGING NETWORKS:

- Drop one key
- All certificates that had been signed by dropped key have to be reissued in the long run
- Simplest case: one network consist of only 1 cluster: CH can be integrated into the other network or dissolved
- Decision on which of the networks is to remain depends on the number of CHs and number of nodes that would have to apply for new certificates
- CHs compete with nodes in existing network to keep their roles by collecting WC.

Details and Protocols

◆ ACCESS CONTROL

- Access to services and resources can be control using authorization certificates
- Entities that are responsible for controlling access can give authorization certificates to the users and allow users to transitively pass those rights to other nodes

Details and Protocols

- ◆ **ADAPTABLE COMPLEXITY:**
 - Two types of keys: symmetric cluster key and asymmetric (network) public key
 - Security levels:
 - . No encryption
 - . Secret cluster key (intra-cluster only)
 - . Public node keys: directly exchanged, certified by CH network
 - Nodes decide on each case which security level is needed and use appropriate encryption
 - If nodes cannot agree to a common level of security then communication is impossible.

Conclusion

- ◆ Distributed infrastructure is highly adapted to the characteristics of Ad Hoc networks
- ◆ Secret sharing avoid central instances that would form single points of attack and failure.
- ◆ Multi-level security model ensure authentication, integrity, and confidentiality.
- ◆ Future improvement:
 - address the impact and limitations of communication technology deployed
 - evaluate time-out for log-on procedure.

References

- ◆ M. Bechler, H.-J. Hof, D. Kraft, F. Pählke, L. Wolf, "*A Cluster-Based Security Architecture for Ad Hoc Networks,*" *IEEE Networks*, March 2004