Simple Key Management for PIM Authentication Keys

Thomas Hardjono Brad Cain

Bay Architecture Laboratory

Nortel Networks

3 Federal Steet

Billerica, MA 01821

USA

{thardjono,bcain}@baynetworks.com

Simple Key Management for PIM Keys

- Key management for a single PIM domain
- Introduce key management entity called *Domain Key Distributor* (DKD)
- The approach relies on limited or "closed" usage of public key cryptography
- Only PIM entities know certain public keys (eg. PK_{dkd} of DKD).
- Notation:

- (PK, SK) denotes Public-Key and Secret-Key pair (asymmetric)
- K denotes symmetric key
- Square brackets [ ] denote digital-signature / authentication (asymmetric/symmetric)
- Curly brackets { } denote encryption (asymmetric/symmetric)
- C is ciphertext

	Assigment of Primary Keys	Manual configuration	Dissemination of PK_{bsr}	Dissemination of K_{rp}	Dissemination of K_{eq}
DKD	K_{eq} PK_{bsr} K_{rp}	(PK_{dkd}, SK_{dkd}) PK_{bsr} (PK_{rpbsr}, SK_{rpbsr})	$[PK_{bsr}]_{SK_{dkd}}$	$\{K_{rp}\}_{SK_{rpbsr}}$	$\{K_{eq}\}_{SK_{dkd}}$
BSR	K_{eq} (PK_{bsr}, SK_{bsr}) K_{rp}	PK_{dkd} (PK_{bsr}, SK_{bsr}) (PK_{rpbsr}, SK_{rpbsr})	(as above)	(as above)	(as above)
CRPs	K_{eq} PK_{bsr} K_{rp}	PK_{dkd}	(as above)	(as above)	(as above)
Other PIM routers	K_{eq} PK_{bsr}	PK_{dkd}	(as above)	Drop Message(?)	(as above)

Rekeying K_{rp}

- Assume DKD generates new key K_{rp2} (Old key is K_{rp1})
- DKD encrypts: $C_{rp} = \{K_{rp2}\}_{SK_{dkd}}$
- DKD further encrypts: $CC_{rp} = \{C_{rp}\}_{K_{rp1}}$
- Unicast CC_{rp} to BSR and RP/CRPs or multicast to special group

Rekeying K_{eq}

- Assume DKD generates new key K_{eq2} (Old key is K_{eq1})
- DKD encrypts: $C_{eq} = \{K_{eq2}\}_{SK_{dkd}}$
- DKD further encrypts: $CC_{eq} = \{C_{eq}\}_{K_{eq1}}$
- Multicast to special group