ERICSSON

Distributed Security Infrastructure

Makan.Pouzandi@Ericsson.ca
Ericsson Research
Open Systems Lab
Montréal – Canada

June 26 , 2002

Agenda

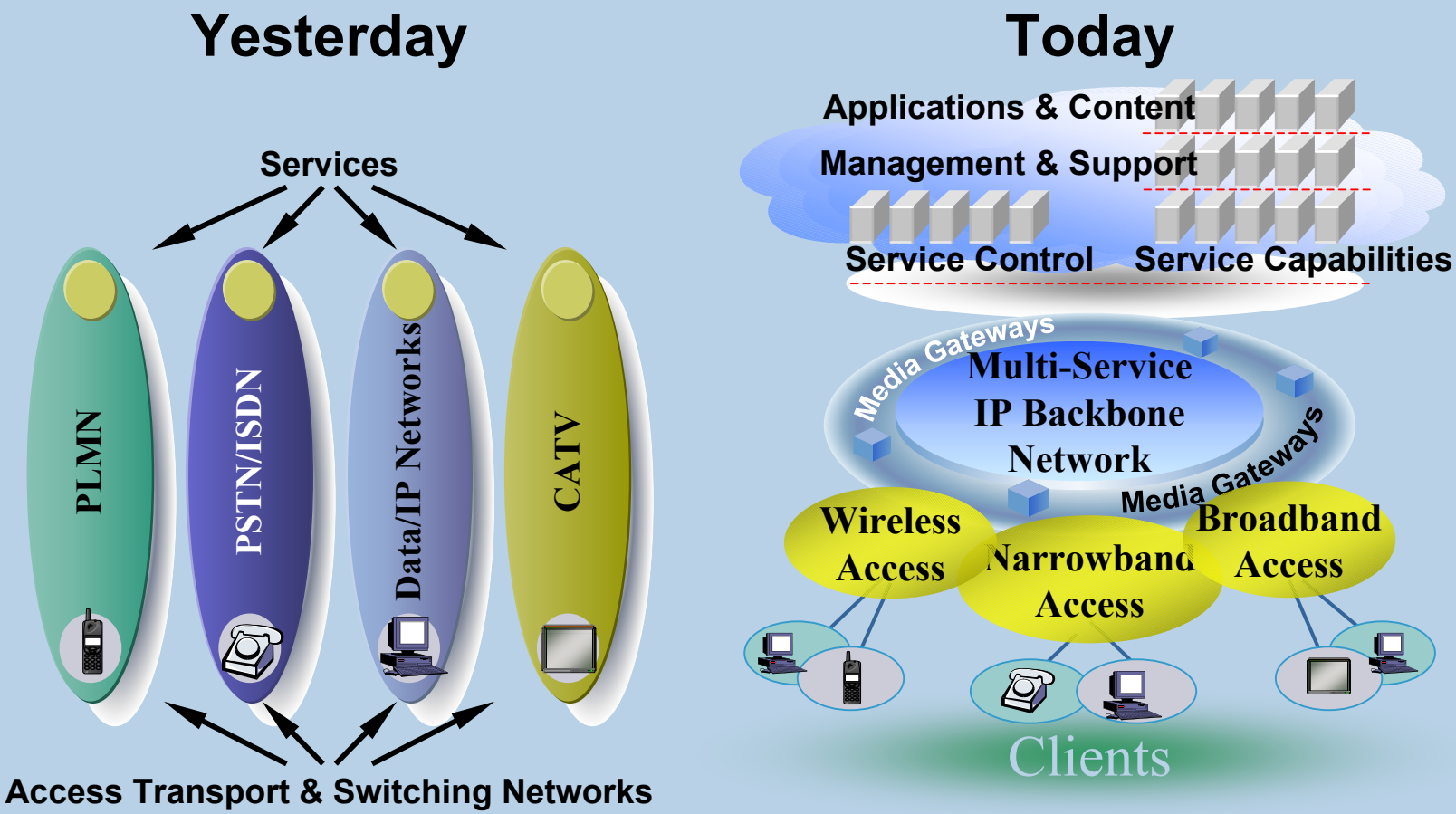
- Context
- Security in Telecom business
- Current situation
- Need for a new software
- DSI Goals and functionality
- DSI overview
- Security services

Context

- Target application: soft real time applications
- High Availability: 99.999% uptime
- Clustered servers
- Exposed to the Internet
- Providing services to different operators
- Running untrusted third-party software
- Software configuration evolves slowly over time: no wild software installations

Why Security in Telecom business?

Change in the market: all-IP-networks



The need for a new approach

“Distributed Systems Require Distributed Security”

Hartman, Flinn, Beznosov,
Enterprise Security with EJB and CORBA

Challenges in Distributed security

- Implement coherent distributed security
  - Many layers to **fit together**: Applications, Middleware, OS, Hardware, Network …
  - Heterogeneous environment: variety of Hardware, Software: OS, Middleware, Networking technologies
- Integration of different security solutions from potentially different vendors …
- System management
  - If manually managed, it may lead to misconfigurations and inconsistencies

Patching versus Coherent framework

- Precise place to intervene when it is necessary to increase performances or the needs for the system change according to the client or legal issues
- Coherent solutions evolve over time; patching does not!

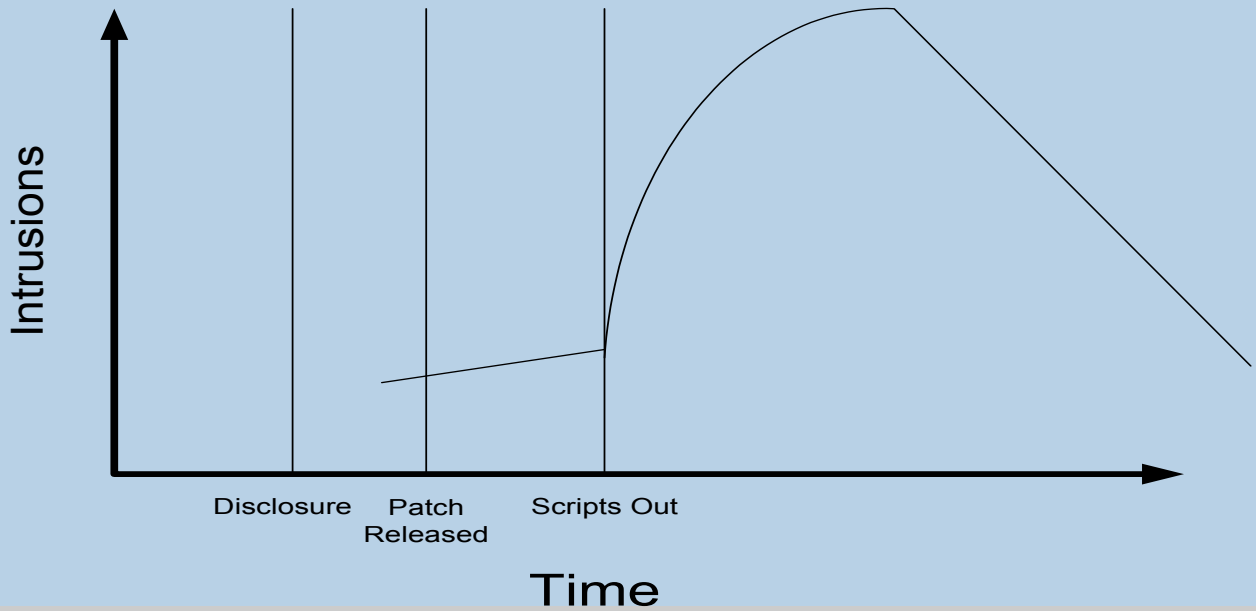


Figure from “Building Secure Software”, Viega-McGraw

Benefits of a coherent framework

- Abstracting the underlying security algorithms and mechanisms
- Reducing development time
- Minimizing the risk of creating subtle, but dangerous security vulnerabilities by reusing security tested software
- Maximize our investment for security mechanisms

Security in different types of Clusters

Traditional Clusters

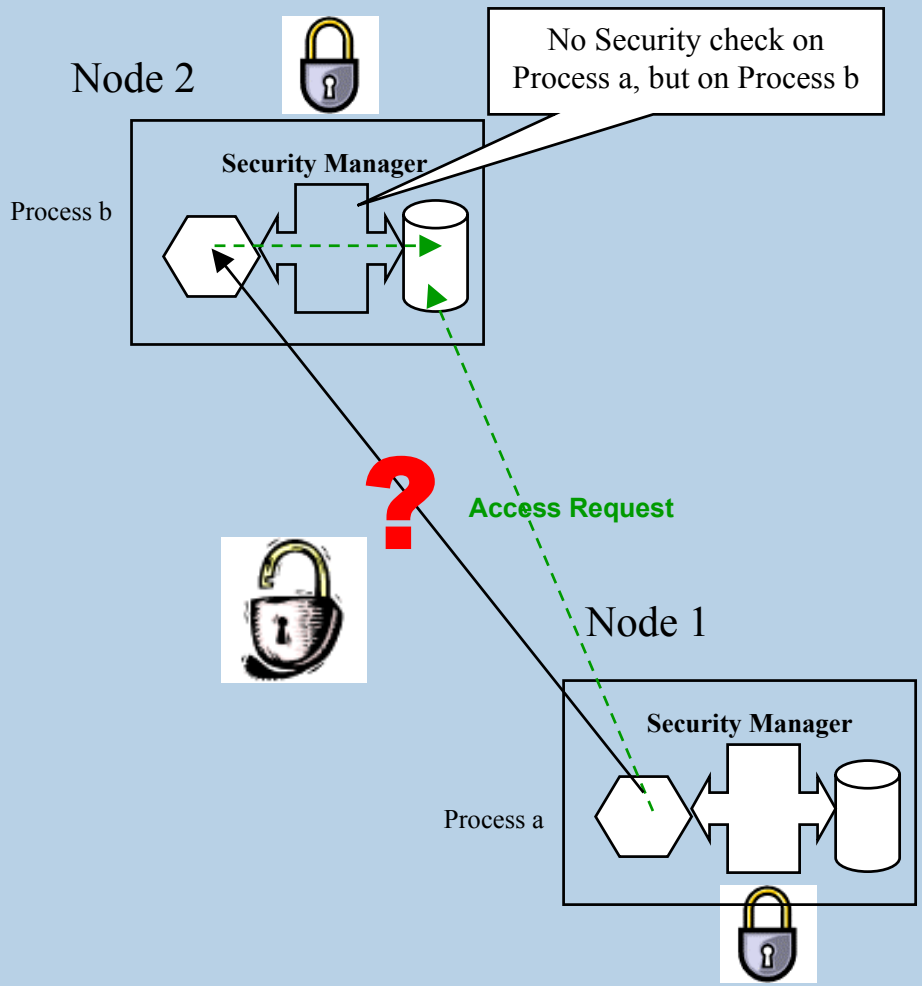
- No real time applications,
- Security policy based upon login and passwords,
- Running for short period of time (days) before each reboot,
- No pre-emptive security.

Carrier Class Clusters

- Target application: soft real time,
- No possible security policy upon traditional login, password,
- Running for a very long time (months) under the same login without rebooting,
- Fine grained security policy based on processes,
- Pre-emptive security.

Access control Approach on cluster computing

- Current security approach in cluster computing:
  - Generally based on user privileges (login, password)
  - Life time: a session of several hours
  - Scope: limited range of operations according to the application's nature
- Our target telecom application:
  - One user only
  - Life time: months if not years
  - Scope: wide range of operations, from upgrading software to managing information in database



Existing solutions

- Many existing security solutions exist:
  - As external security mechanisms to the servers such as firewalls and Intrusion Detection Systems
  - As part of servers such as Integrity checks and some mechanisms to enhance security as a part of OS…

- However, there are few efforts to make a coherent framework for enhancing security in a distributed system

Distributed Security Infrastructure Goals and Functionality

Project Goals

- Design an architecture that:
  - Supports security mechanisms to protect the system against External attacks originating from Internet, Internal attacks (Break through a node in the cluster, O&M security, Intranet attacks ..)
  - Accommodates current and future needs
  - Provides mechanisms for detecting and reacting to breaches
  - Targets Carrier Class ***Clustered*** Server
- Architectural Requirements:
  - Scalable and Flexible
  - Does not provide a single point of failure
  - Does not impose any performance bottlenecks
  - Provide ease of development

DSI characteristics

- Coherent framework: coherent through different layers of heterogeneous hardware, applications….
- Process level approach: security based on individual processes
- Pre-emptive security:changes in the security context will be reflected immediately
- Transparent key management: cryptographic keys ecurely stored and managed
- Dynamic security policy: run time changes in security context and policy

What we do vs. what we don't do

Do

- Design and implement a coherent framework for the security needs of a cluster running a soft real time application
- Re-use as much as possible existing algorithms and protocols (COTS)
- Adapt current technologies to fit our needs and environment (soft real time)

Do Not

- Invent new algorithms nor new protocols for cryptography, authentication or else

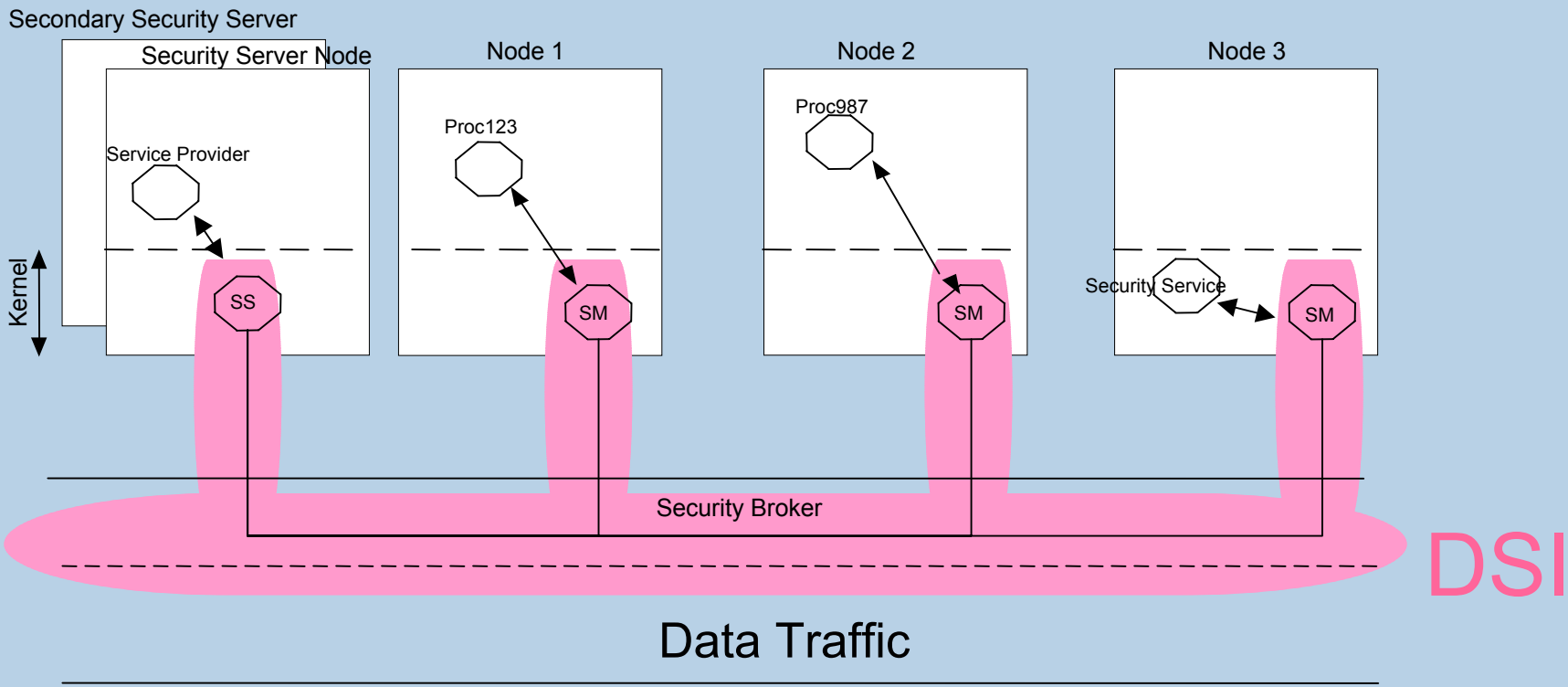
DSI Functionality

- Access control: resources each subject should be able to access and prevent the illegal accesses
- Authentication: verifies that the principals are who they claim to be.
- Auditing: provides a record of security relevant and allows monitoring of the subject in the system.
- Confidentiality and Integrity for communications
- Security Management

ERICSSON

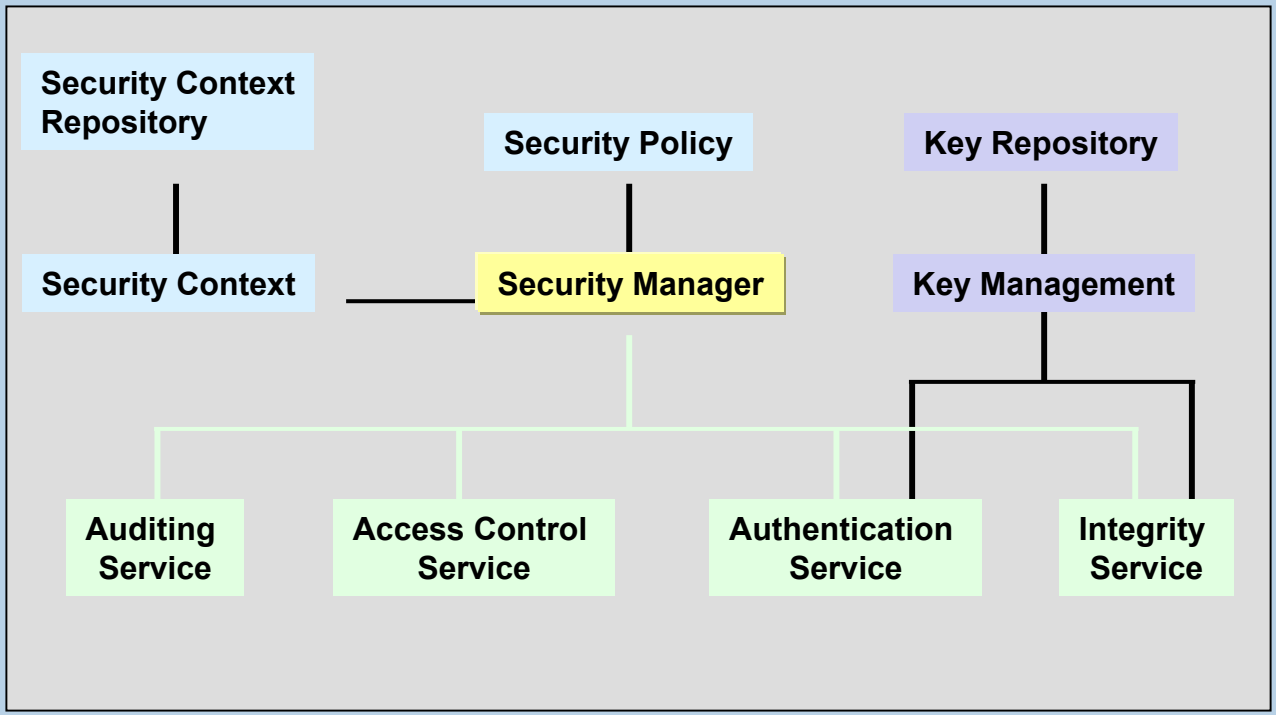
Distributed Security Infrastructure Overview

Distributed Architecture



SM: Security Manager
SS: Security Server

Security Services



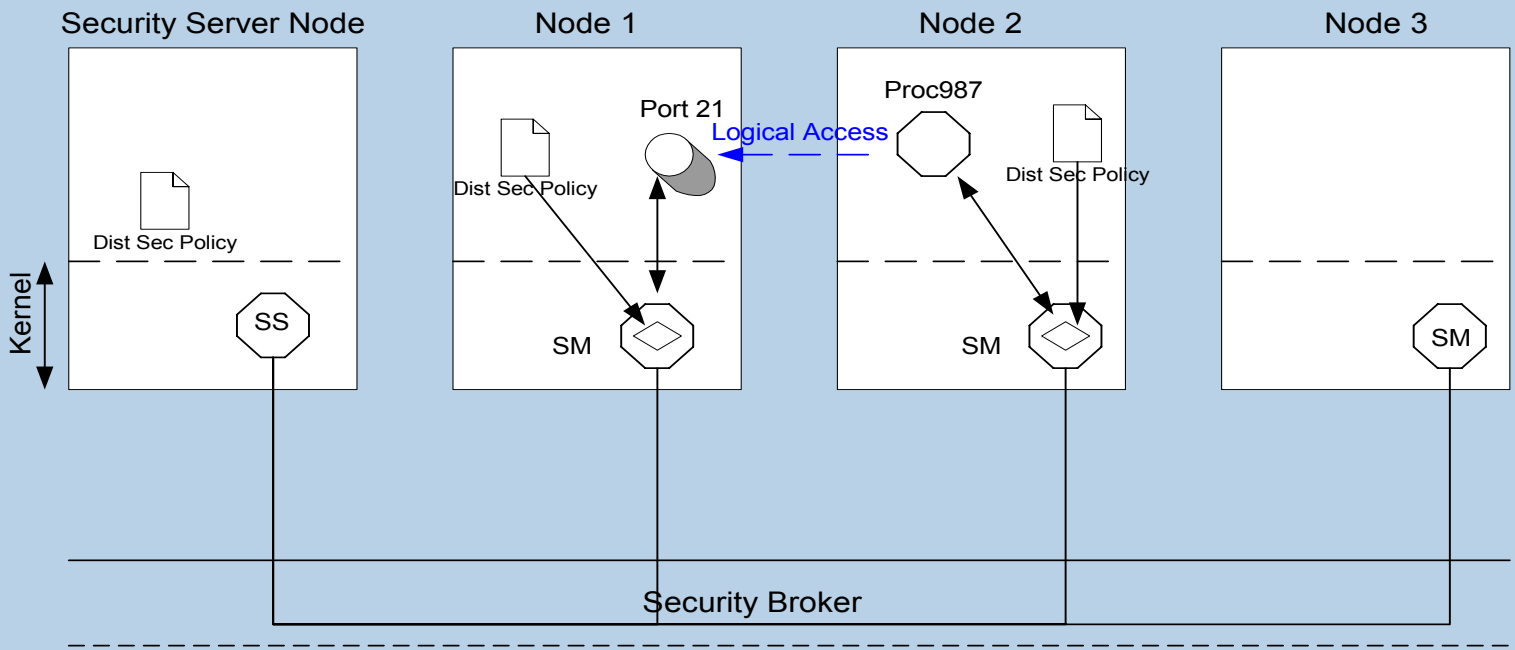
Service based (2)

- Separation between API and Implementation
  - Implementation changes, security patches do not affect the system
- Flexibility
  - Easily change, update, remove services based on needs, legal issues
- Evolution over time

Distributed Security Policy (DSP)

- Express a coherent security vision (security policy) through out all the cluster
- Local security policy:
  - Initially integrated to the secure boot software
  - Maintained and updated by the security server through security broker
- Based on domain enforcement
- Define communication type between processes: secure, not secure, authenticated, encrypted…

Distributed Security Policy



Data Traffic

SS: Security Server
SM: Security Manager

DSI Core

Security Server, Security Manager and Security Communication Channel

Development Environment

- Kernel 2.4.17
- LSM patch 2.4.18
- Red Hat 7.2
- C/C++
- GCC 2.96

Secure Boot

- Secure Boot: provides us with Distributed Trusted Computing Base (DTCB)
- Kernel at secure boot is small enough to be thoroughly vulnerability tested
- Use of digital signatures and a local certification authority will prevent DTCB from malicious modifications

Secure Boot Status

- Development software kit done
- Download boot images from the network
- Checks RSA signatures on boot images
- Executes the boot image
- Kit based on
  - Network-Boot kit
    - boots from LAN
    - runs Linux
    - diskless (RAM based)
  - Two-kernel Monte
  - OpenSSL 0.9.5

Security Server

- Security server is the reference for all security managers
  - It can declare a node compromised
- It manages the following tasks:
  - Monitoring:
    - Audit the cluster: Testing the heart beats from Security managers, sending challenges to check their authenticity
    - Audit the internal sub network between nodes for detecting attacks or intrusions,
  - Triggering alarms, warnings to inside and outside of the cluster
  - Distributed Security Management
  - Propagate security related info through security broker: Distributed Security Policy Updates, Node security status, Alarms, Warnings
- Entry point to the DSI for administrators

Security Server Status

- Event Driven approach: Handlers/Callbacks for different types of events
  - More efficient and responsive to events (incidents)
  - Less resource consuming, can run on background and only wake up when receiving events
- Threads for each type of channel of SCC
- Scheduler thread to trigger regular audits, heart beats, security checks

Security Server Status (2)

- Focus:
  - Event driven architecture
  - Triggering alarms, updates…
  - Entry point for admins,
  - Basic GUI for displaying alarms, warnings,… from SMs: GTK 2.0
- First prototype done

Security Manager

- Enhance security mechanisms locally for a node,
- It manages the following tasks:
  - *Key Management*: generating, storing, and retrieving keys for local processes
  - Make access control decisions
  - Authenticate local and remote processes
  - Ensure the integrity of data sent and received
  - Security Context management
    - Assigns SIDs for local entities
    - Caching and cache coherency for security contexts
  - Vehicle the security status through a security broker to the security server

Security Manager Status

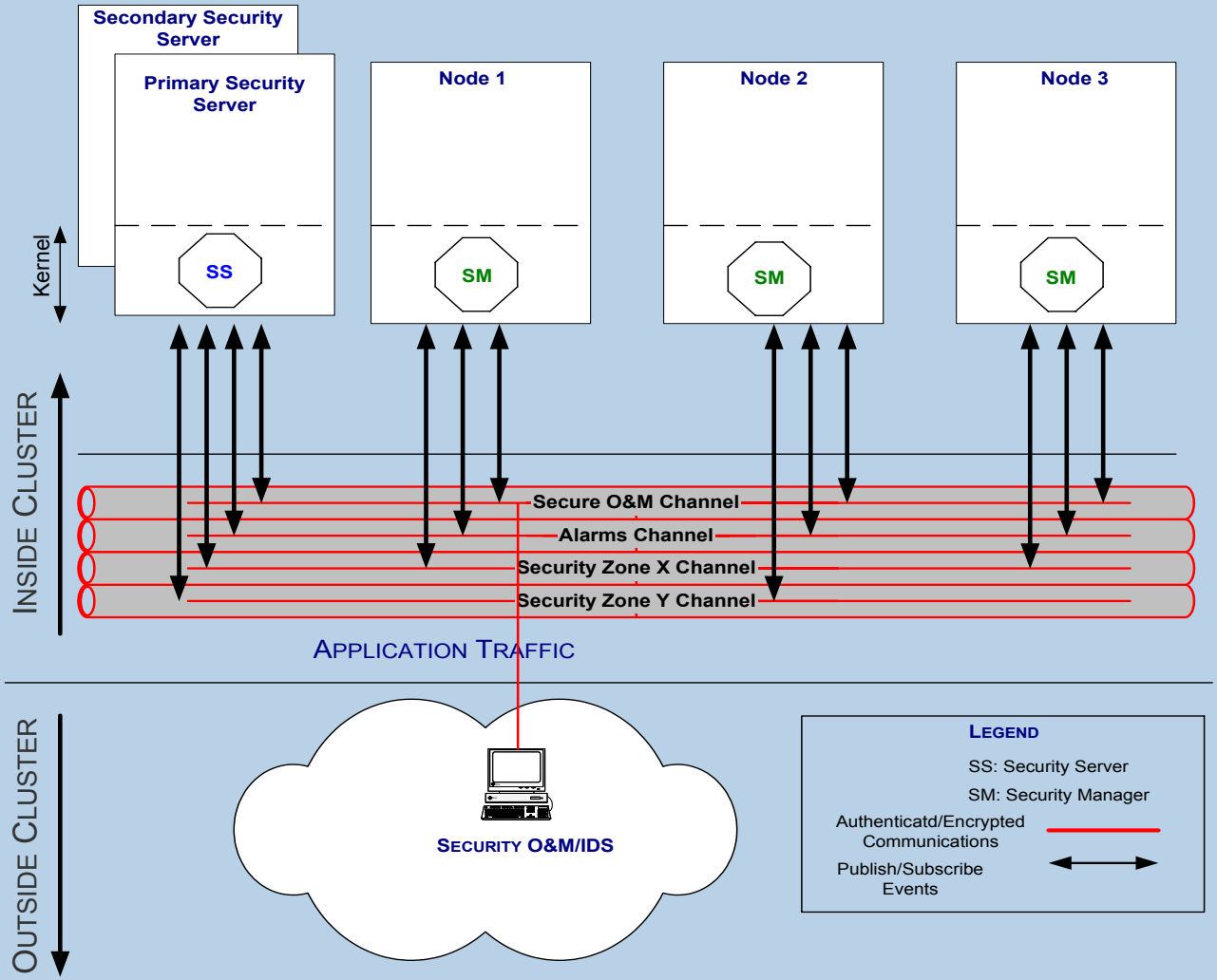
- Event Driven Approach, idem to Security Server
- Threads by types of channel

- Connect to DSM through system calls

- Focus:
  - Security Info sent and received from SS
  - Interfacing with DSM

- First prototype done

Secure Communication Channel (SCC)



SCC Functionality

- Broke security related info to all security elements
  - Attention: SCC is not used for application data but security related information
- All communications authenticated and encrypted
- Based on communication channels, Publish/Subscribe approach
- Portability layer
  - Published API to services must be independent from underlying security mechanisms
- Priority queuing
- Based on known standard security protocols

Advantages of using event channels

- No single point of failure
- Inherent event filtering
  - Less network bandwidth
  - Less CPU and memory for discarding irrelevant messages

SCC Status

- Based on CORBA
  - Omni ORB 3.0.5
  - Why CORBA has been chosen ?
    - Support for Distributed Real time and embedded systems
    - Support for Advanced security mechanisms: CORBA SEC
    - Interoperability
- XML used for Messages and Commands
  - Xerces 1.7.0
  - XML1.0, SAX 1.0
  - Why XML has been chosen:
    - Self Described messages: easily augment the language
    - XML has its own mechanisms for security: digital signature, extra security when paranoid situation
    - Comes with free parsers

SCC Status (2)

- Focus
  - Portability layer done
  - Publisher/Subscriber layer for Event service done
  - Channel management logics done
  - Logics done

ERICSSON

DSI Security Services

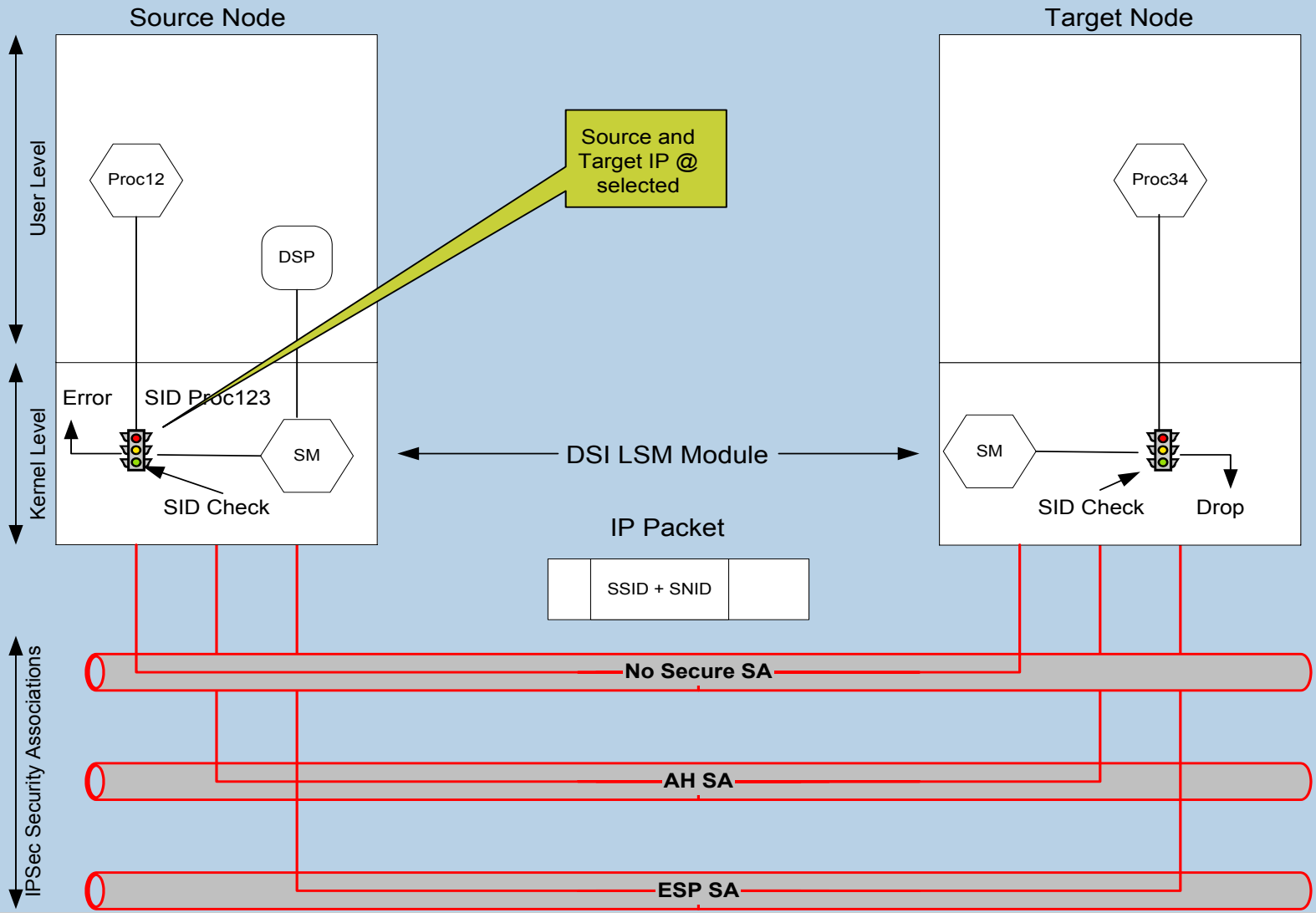
Services

- Access Control Service
- Authentication Service
- Integrity and Confidentiality for communications service
- Auditing Service

Integrity and confidentiality for communications (ICC)

- Based on IPSec
- Security Association chosen based on SID of initiating process and DSP
- 3 kinds of SA:
  - AH
  - ESP
  - No Security
- Different IP addresses are used for each SA
- DSM choose source and destination IP according to SID and DSP

IPSec and DSI



Advantages

- Fine grain control
  - Type of encryption chosen according to the process
- Flexibility: Modifying dynamically the security parameters according to the context: load, security incidents,…
- Security Administrator manages the SA type to be used
  - Enhance security for third-party software,
- Transparent to application
  - Useful for third-party software
- Enhancing security by firewalling rules for each sub network used for each SA type: AH, ESP, No

ICC Status

- IPSec implementation FreeS/WAN 1.94
- No modification to the kernel
- IP source and destination addresses passed to DSM through LSM hooks
- DSM change addresses based on SID and destination IP only
- First prototype done
  - Running with FreeS/WAN
  - Problems with use of IP Options and FreeS/WAN

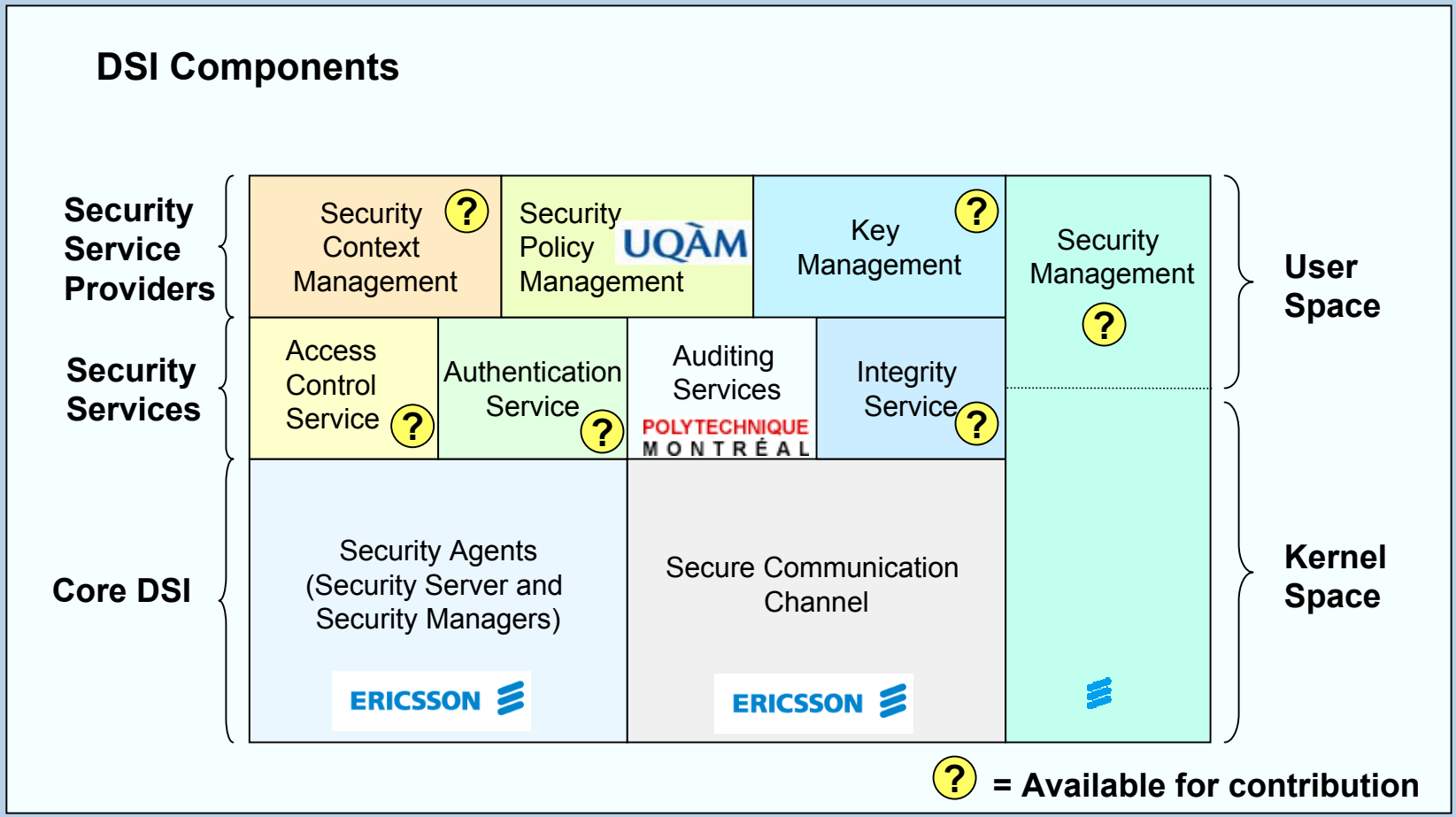
ERICSSON

DSI as an Open Source Project

Why DSI an Open Source Project?

- Get proper peer review
- React fast in a full-disclosure world
- Establish common framework
- Joining our forces to do more and better

Different DSI Components



Project Status

- Implemented:
  - Distributed Trusted Computing Base (DTCB): secure boot mechanism for a diskless Linux
  - DSI Linux Security Module
  - SCC based on OmniORB, Open Source CORBA implementation
- On going
  - Core DSI: First implementation done
  - DSI Authentication and Integrity service based on DSM and IPSec: First prototype done
  - Integration of DSM and SCC: Distributed Access Control to be extended to all necessary operations in the cluster
  - Work on DSP

Conclusion

DSI is important because …

- DSI provides with:
  - Protection against security attacks
  - In the case of security breach:
    - Efficient Mechanisms for Detection
    - Fast Reaction to control damage

ERICSSON

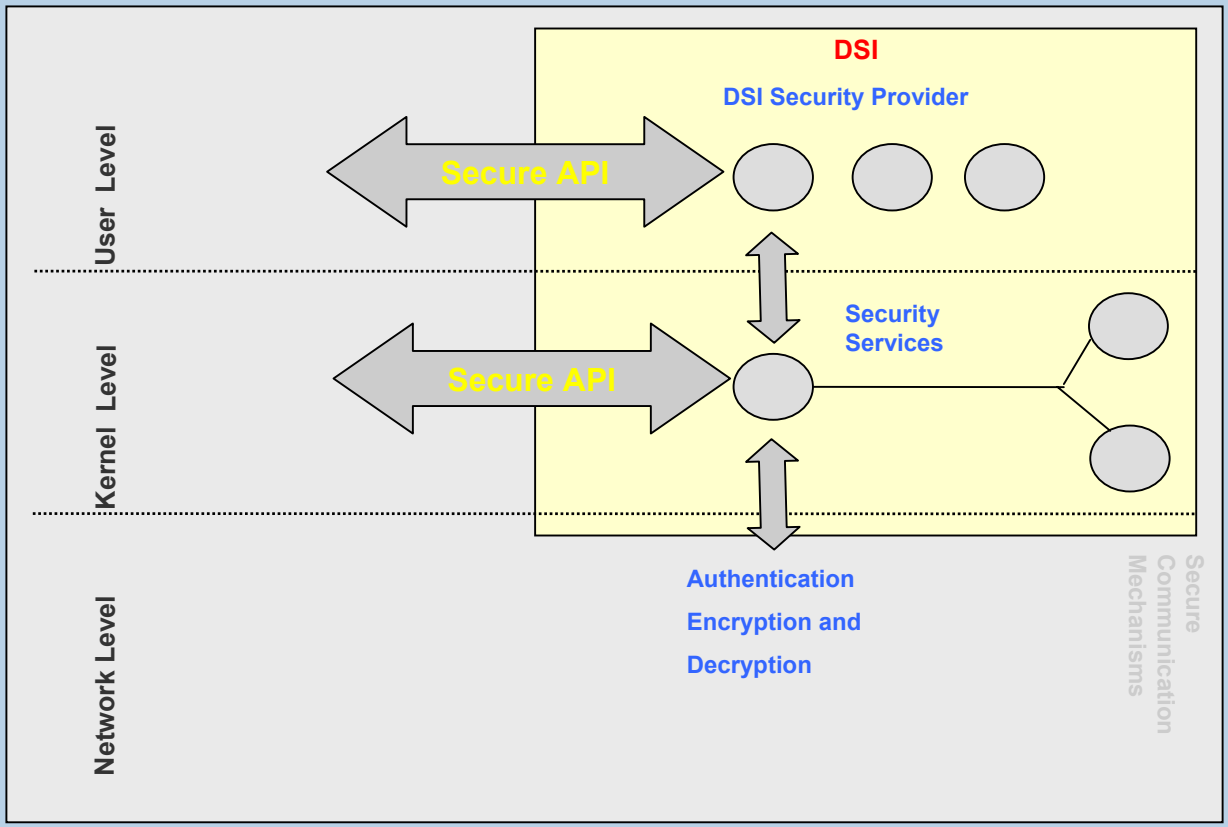
DSI is Open Source and we need contributors

- Feedback
- Contribute some work
- Become partner

Contact Info

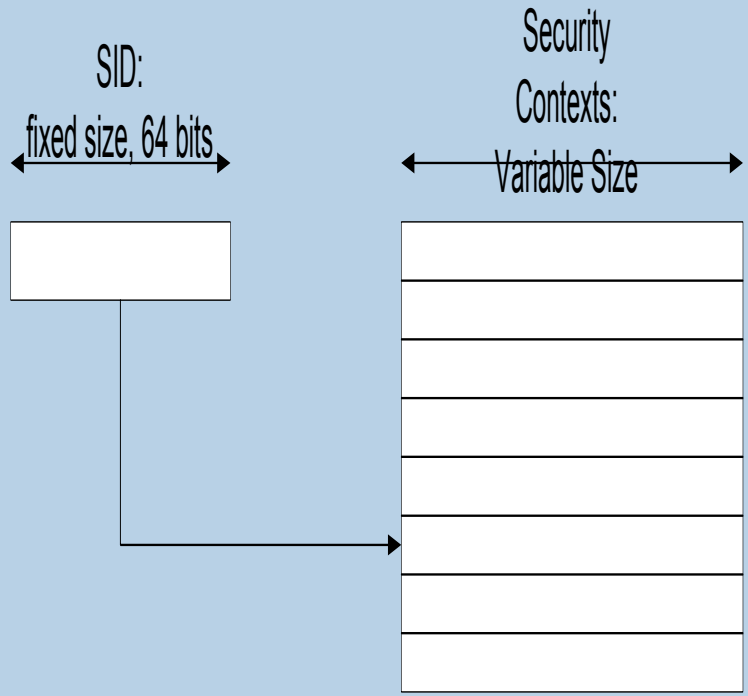
- Email: Makan.Pourzandi@Ericsson.Ca
- Web Site: http://www.risq.ericsson.ca (206.167.214.56)

DSI inside a node: layered approach



Security context

- Privileges associated with each process, defined through the whole cluster
- Security ID: fixed size value corresponding to the security context
  - Can be transferred and interpreted through the whole cluster,
  - Assigned by local security manager,
  - Unique for each entity in the distributed system,



Telecom business changes…

- Change in the market: all-IP-networks
- Increasing number of attacks via the Internet
- Huge demand for security

Some facts on security

- Security is a chain; it is only as secure as its weakest link. “Schneier”
- There is no 100% secure systems
- Security based on firewalls is not enough to stop hackers: Hard outside, Soft inside
- Defence in Depth

Authentication service

- Fine grained authentication: process
- Local authentication: Based on verification by DSM at kernel level
- Remote authentication: Local authentication extended by the use of IPSec
  - IPSec provides secure sessions between nodes with
    - Authentication
    - Data integrity and confidentiality

Auditing Service

- Defined requirements
- Based on Open source project EVLOG API

Increasing number of attacks via the Internet

- 4,000 denial-of-service attacks every week
  (University of San Diego researchers, June 2001)

- Organizations victim of attacks via the Internet increased from 38 percent in the 1996 survey to 70 percent in 2001
  (2001 Computer Crime and Security Survey)

More money is spent on secure platforms and applications

- Companies will spend 4% of their revenues on information security in 2011, up from 0.4 percent in 2001

  (Gartner Institute)

- Gartner analyst firm Dataquest forecast that the worldwide security-software market will grow to $4.3 billion this year, up 18 percent from $3.6 billion in 2001. Meanwhile, managed security services should grow even faster, according to market researcher IDC, which estimates that such network-protection providers will take in $2.2 billion in 2005, up from $720 million in 2000.