

SNMPv3

Prof. Mauro Tapajós



SNMPv3

- Não é toda uma nova versão, e sim um complemento para as atuais versões do SNMP
- Pretende oferecer serviços robustos de segurança, infra-estrutura administrativa e configuração remota de agentes
- Possui um formato distinto para a mensagem SNMP, mas prevê que agentes e gerentes SNMPv3 possam se comunicar com entidades SNMPv1 e SNMPv2 também
- É um padrão IETF (Standard 62 - RFC's de 3411 a 3417)

Documentação SNMPv3

- Especificações divididas em vários documentos, numa estrutura modular permitindo evoluções independentes de cada módulo
- Várias novas MIB's definidas
- Definições SNMPv3:
 - Módulos da MIB;
 - Operações do Protocolo e
 - Segurança e Administração
- SNMPv3 = SNMPv2c + infra estrutura de Segurança e Administração da arquitetura de gerenciamento

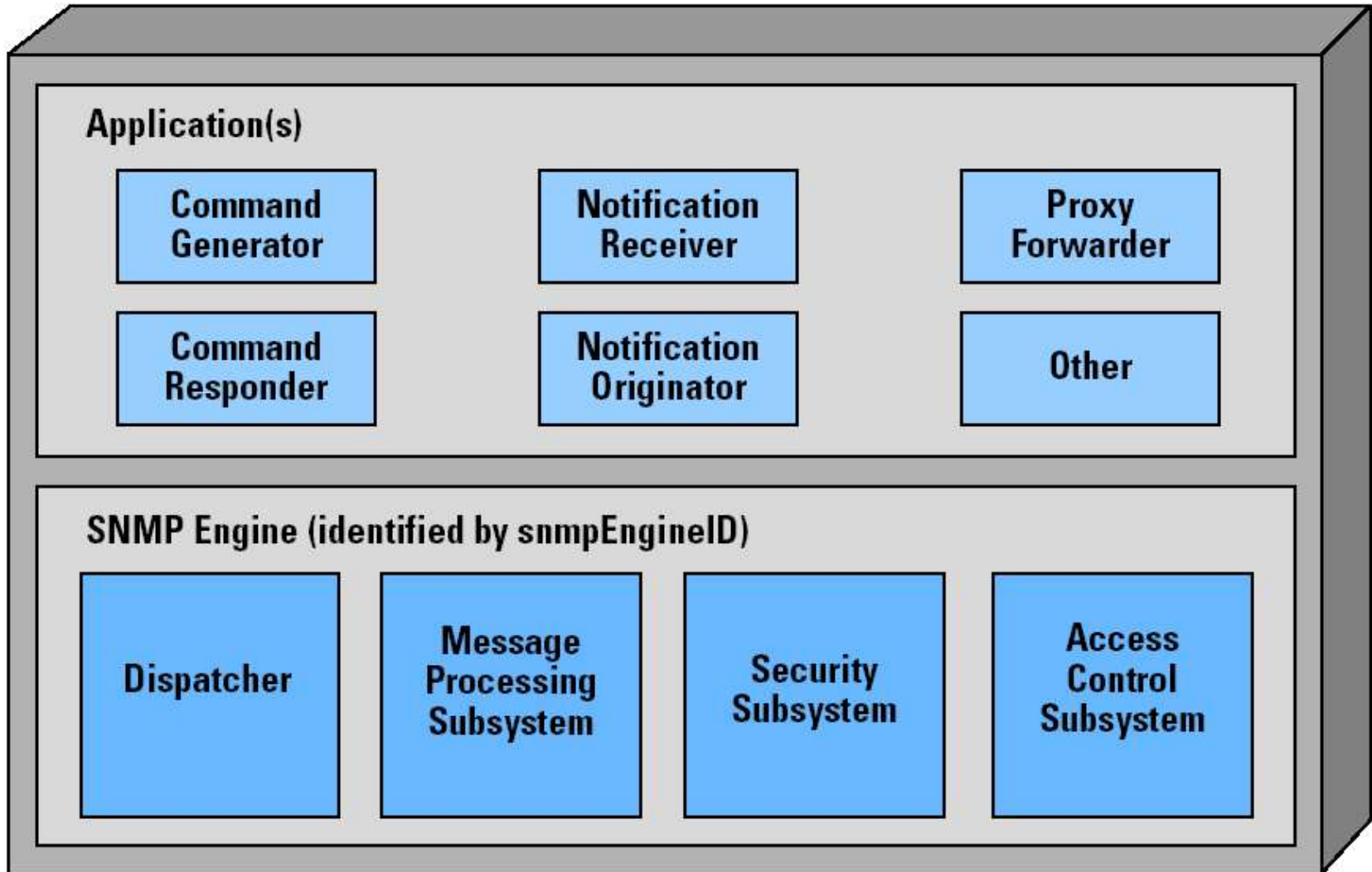
Nova versão SNMPv3

- Mesmos componentes e arquitetura das versões anteriores – Gerentes, agentes e MIB's
- Documentos introdutórios (interessante a leitura!)
 - RFC 3410, “Introduction to version 3 of the Internet-Standard Network Management Framework”, uma visão geral do SNMPv3 e um *roadmap* para os demais documentos;
 - RFC 3411, “An Architecture for Describing SNMP Management Framework”, descreve a arquitetura do protocolo, como um todo, dando especial ênfase aos modelos de segurança e administrativo

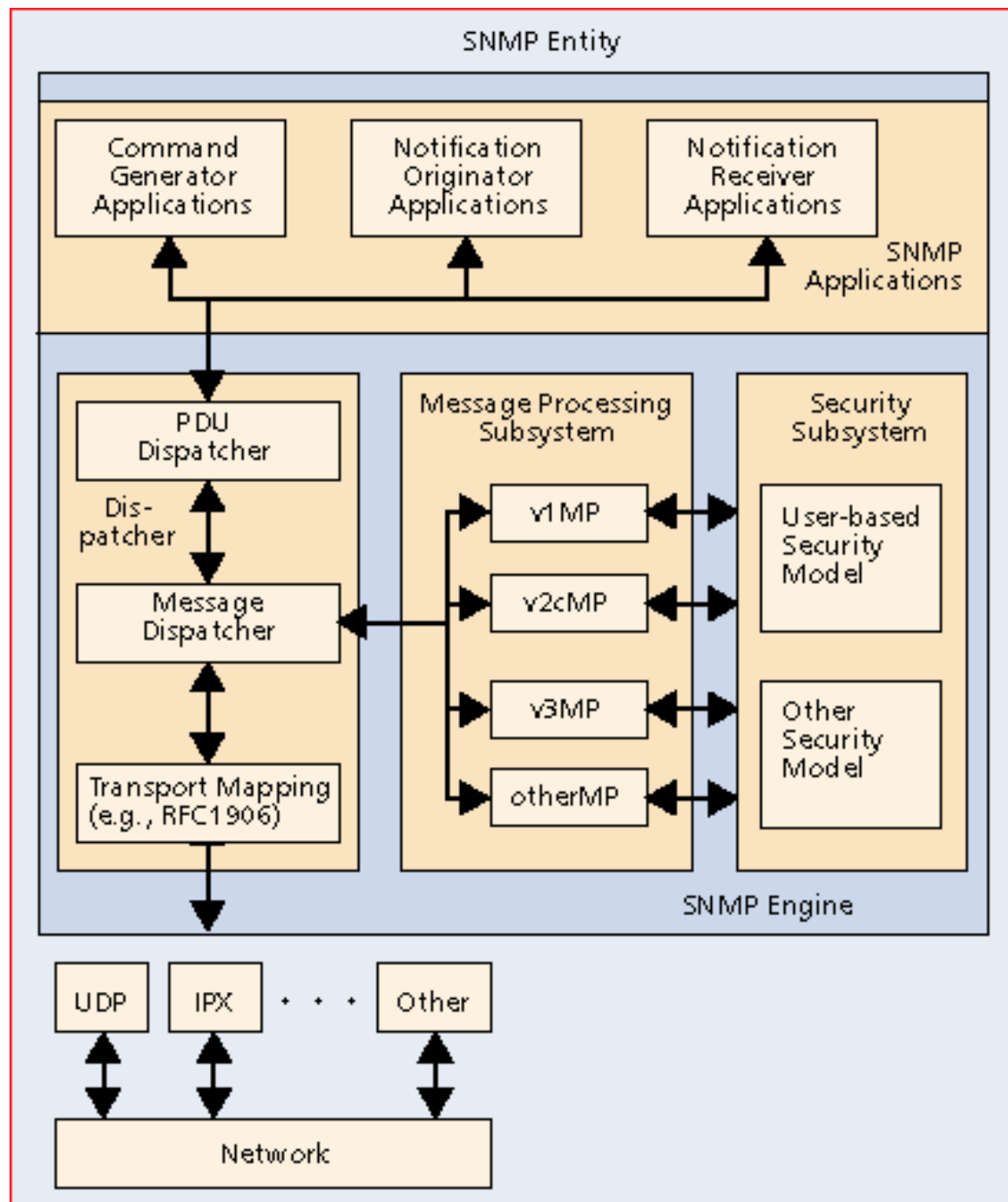
Arquitetura SNMPv3

- Compostas por módulos que interagem provendo serviços uns aos outros, através de primitivas
- Conceitos
 - **Entidades SNMP** (*SNMP Entities*)
 - **Contextos SNMP** – subconjunto definido da informação de gerenciamento
 - **SNMP Engine** (“máquina virtual” que realiza o envio/recepção, encriptação/decriptação de mensagens) – funcionalidades básicas
 - **Aplicações SNMPv3** (*command generator, notification originator, notification receiver, command responder, notification originator, proxy forwarder*) – funcionalidades específicas
- As engines provêem serviços para as aplicações;
- Entidade = Engine + Aplicações

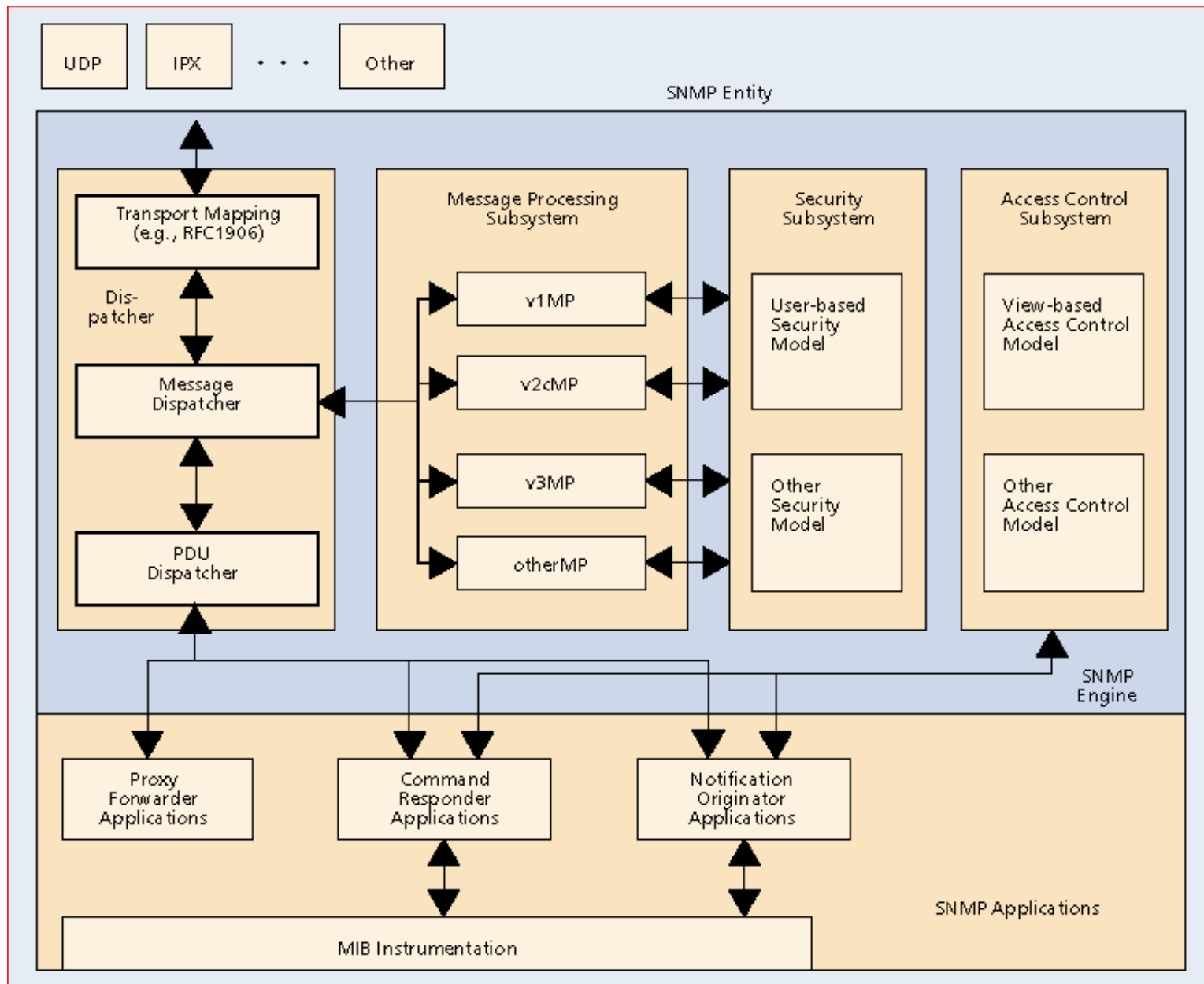
Arquitetura Modular SNMPv3



Arquitetura Gerente SNMPv3



Arquitetura Agente SNMPv3



Funcionalidades SNMPv3

- O processamento da mensagem SNMPv3 é executado pelas funções do *Dispatcher* e do *Message Processing Subsystem*, ambos parte do engine SNMP
- O modelo de segurança é responsável pelo processamento de segurança da mensagem, implementando mecanismos para garantir a segurança da mensagem
- O único modelo de segurança proposto até o momento para SNMPv3 é o modelo USM (*user based security model*) mas nada impede a oferta de outros modelos de segurança no futuro

Segurança SNMPv3

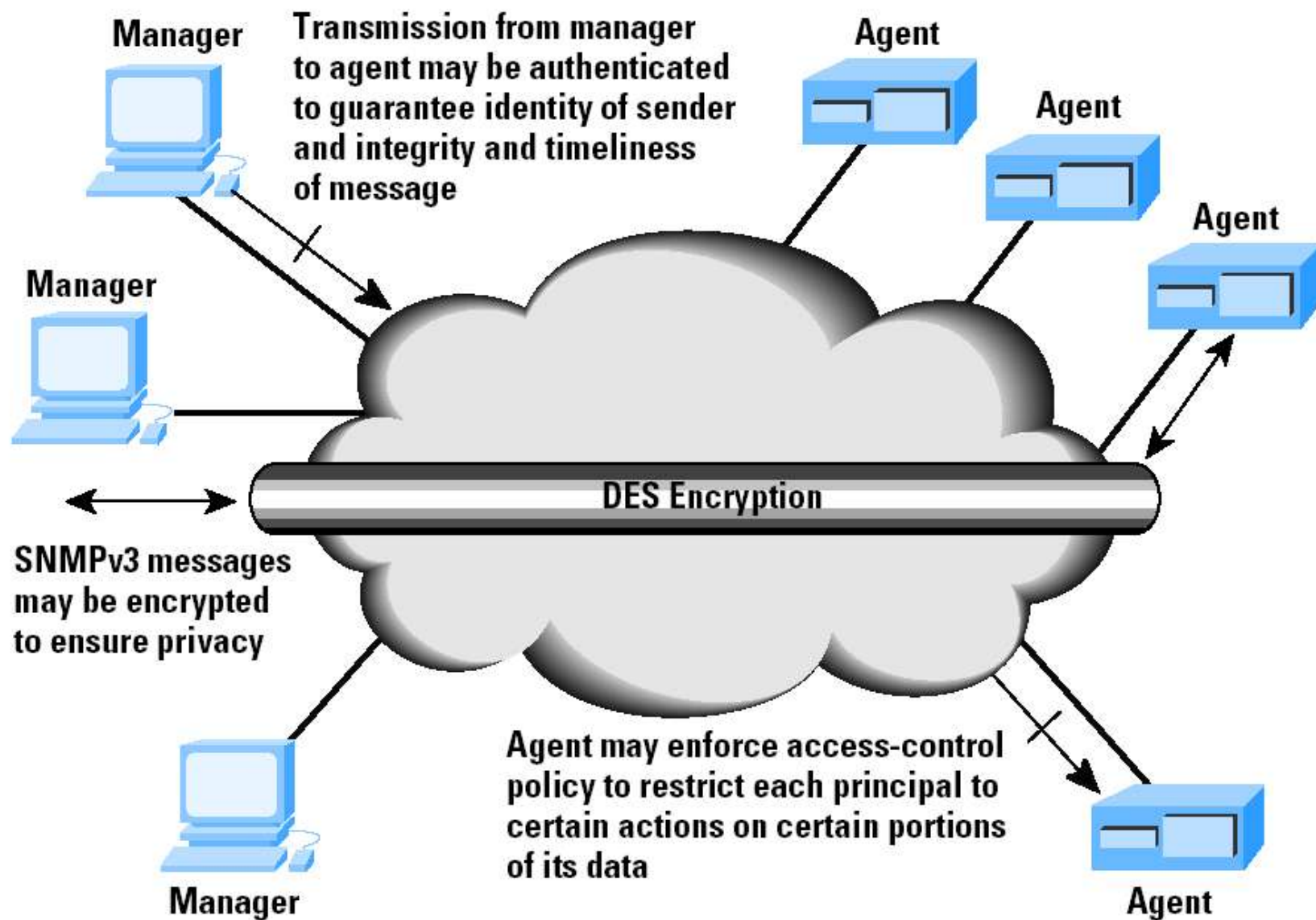
Documentos relacionados

- RFC 3414, “The User-Based Security Model for Version 3 of the Simple Network Management Protocol (SNMPv3)”, descreve ameaças, mecanismos, algoritmos, serviços de segurança e tipos de dados usados com o objetivo de prover segurança ao protocolo;
- RFC 3415, “View-Based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)” descreve como é definido o controle de acesso às informações de gerenciamento;

Segurança SNMPv3 – Ameaças Consideradas

THREAT	ADDRESSED?	MECHANISM
REPLAY	YES	TIMESTAMP
MASQUERADE	YES	MD5 / SHA-1
INTEGRITY	YES	(MD5 / SHA-1)
DISCLOSURE	YES	DES
DENIAL OF SERVICE	No	
TRAFFIC ANALYSIS	No	

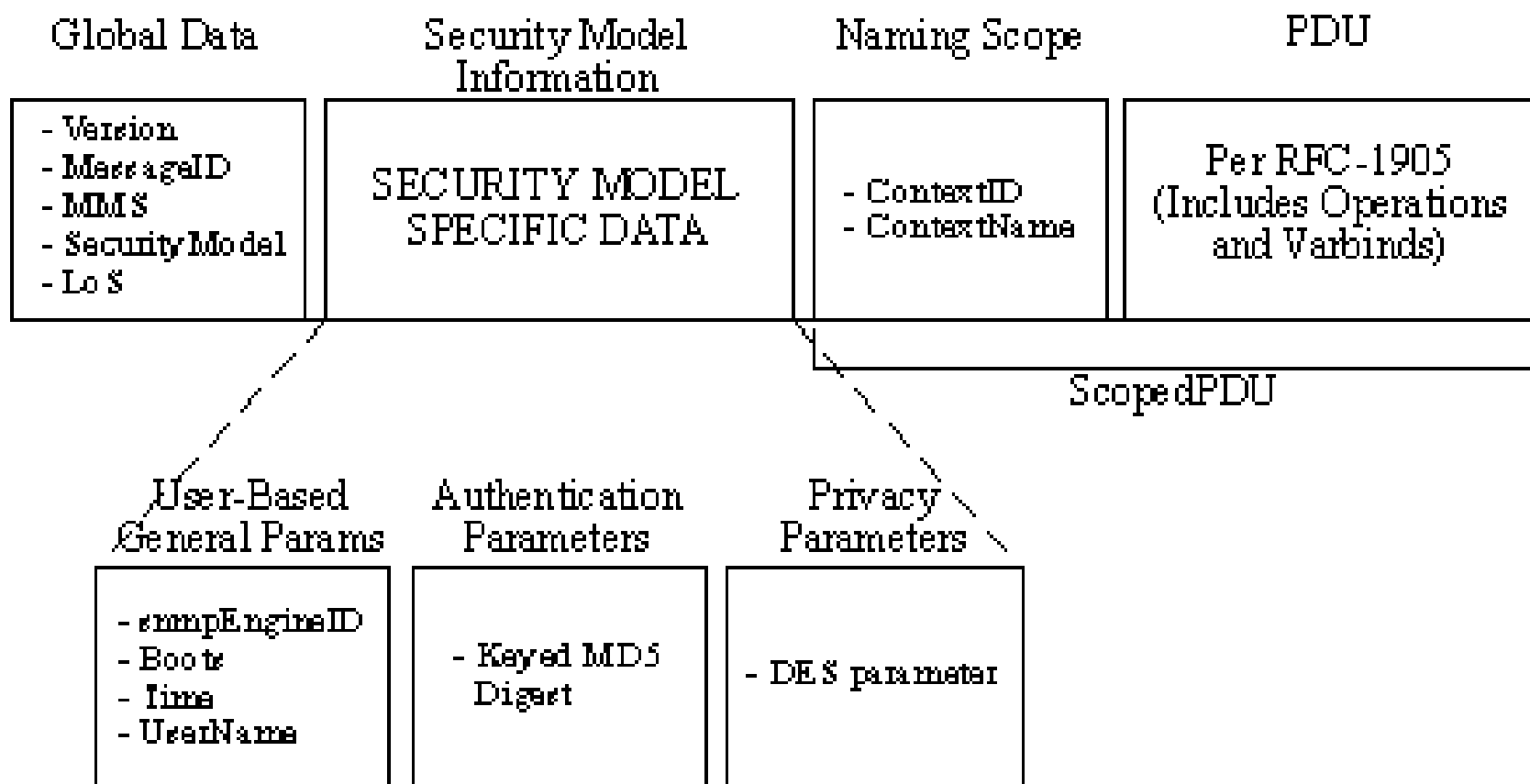
Recursos de Segurança em SNMPv3



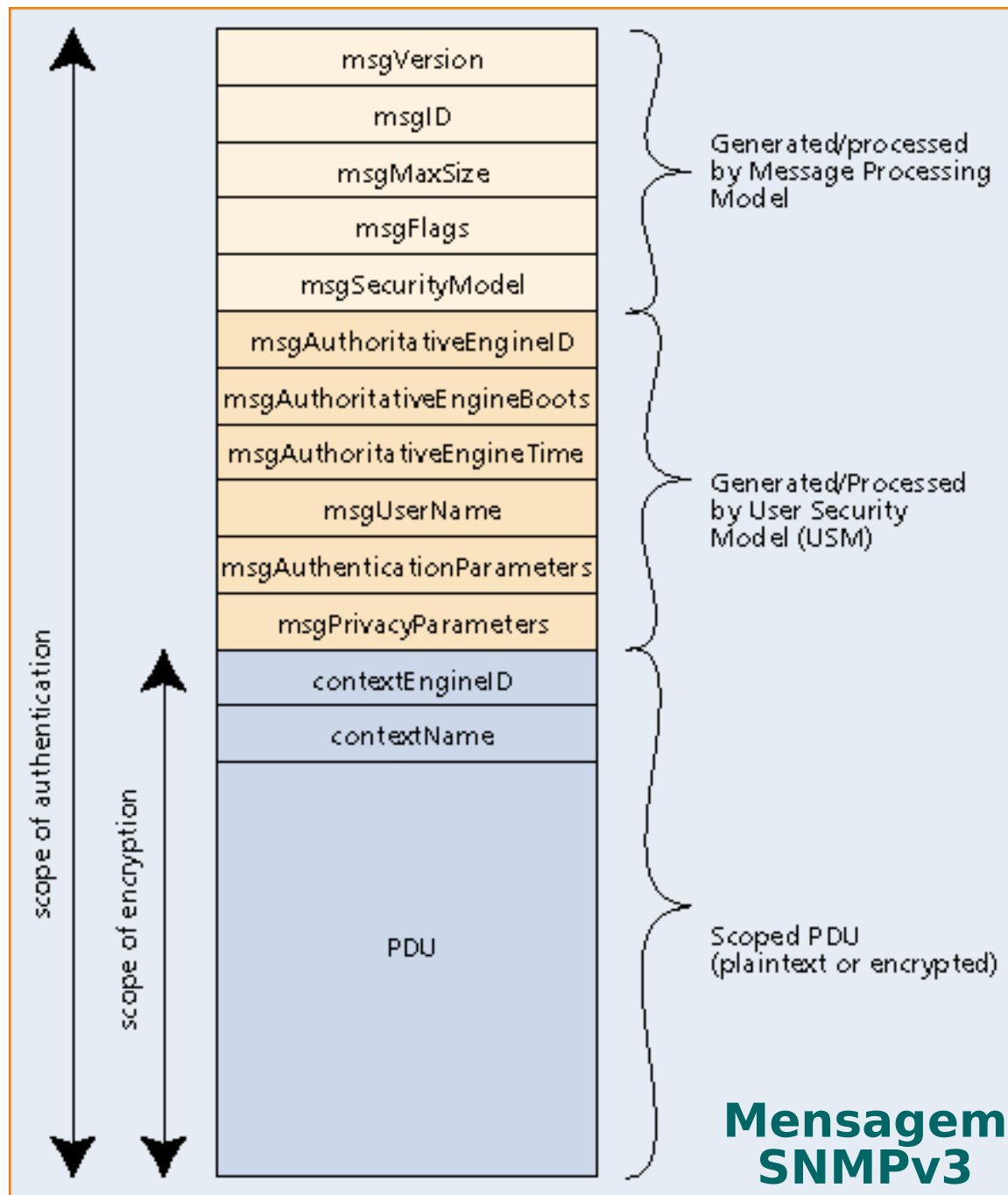
Identificação e Privilégios no modelo USM

- ***Principal*** – uma entidade que pode ter serviços ou processamento SNMP sendo executados. Pode ser um indivíduo, um grupo de indivíduos, uma aplicação ou combinações destes.
- Um *principal* tem vinculadas várias informações de segurança, inclusive as chaves criptográficas e contextos possíveis
- A associação de um principal com um sistema agente define os aspectos de segurança que serão invocados (autenticação, privacidade e controle de acesso)
- O modelo oferece os seguintes serviços:
 1. **Autenticação:** Protocolos HMAC-MD5-96 e HMAC-SHA-96
 2. **Privacidade:** Protocolo DES em modo CBC
 3. **Proteção contra atrasos e reenvios** de mensagens através de mecanismos de temporização

Formato da Mensagem SNMPv3



- **msgVersion** – Especifica a versão do protocolo em uso (3 - SNMPv3)
- **msgID** – Identificador usado para coordenar requests e responses
- **msgMaxSize** – Informa o maior tamanho de mensagem suportado
- **msgFlags** – String que identifica a existência de report, autenticação e encriptação na mensagem
- **msgSecurityModel** – Identifica o modelo de segurança em uso
- **msgSecurityParameters** – String com parâmetros a serem processados pelo Security Subsystem (depende do modelo de segurança)
- **contextEngineID** – Identifica que aplicação o PDU está relacionado
- **contextName** – identificador do contexto sendo usado pelo pdu
- **Scoped PDU** – É o PDU SNMPv2



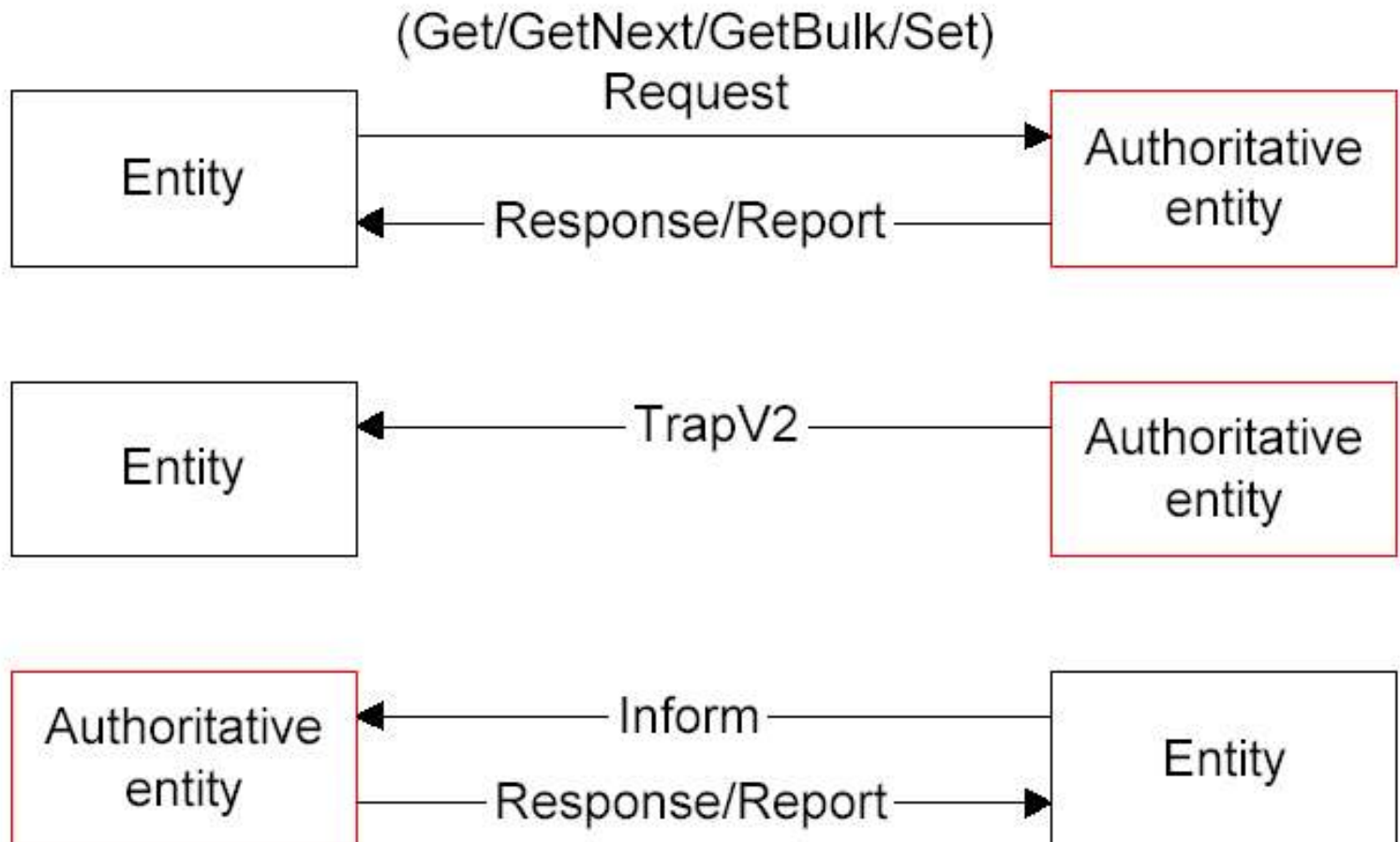
Flexibilidade com Modelos de Segurança

- No futuro, novos algoritmos poderão ser usados sem alterações no modelo
- Seu único protocolo de privacidade sendo proposto é baseado no algoritmo DES
- Ao se enviar uma mensagem pode-se optar por uma das seguintes opções (campo *msgFlags*):
 - sem autenticação, sem privacidade
 - com autenticação, sem privacidade
 - com autenticação, com privacidade
- Neste modelo, autentica-se um usuário (*principal*)

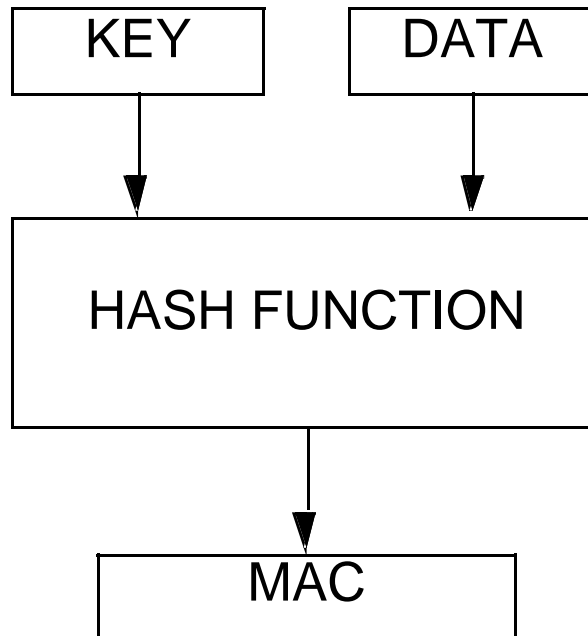
Sincronização de Mensagens em USM

- As mensagens são protegidas por um mecanismos de sincronização para evitar ataques de reenvio de mensagens já enviadas
- Este mecanismos é baseado no conceito de *Authoritative SNMP engine*, usado para definir a noção de tempo correta para uma mensagem
- Os campos *msgAuthoritativeEngineBoots* e *msgAuthoritativeEngineTime*, dão a indicação de tempo necessária para um serviço de autenticação de um *engine* definir se uma mensagem é “velha” demais

Sincronização de Mensagens em USM

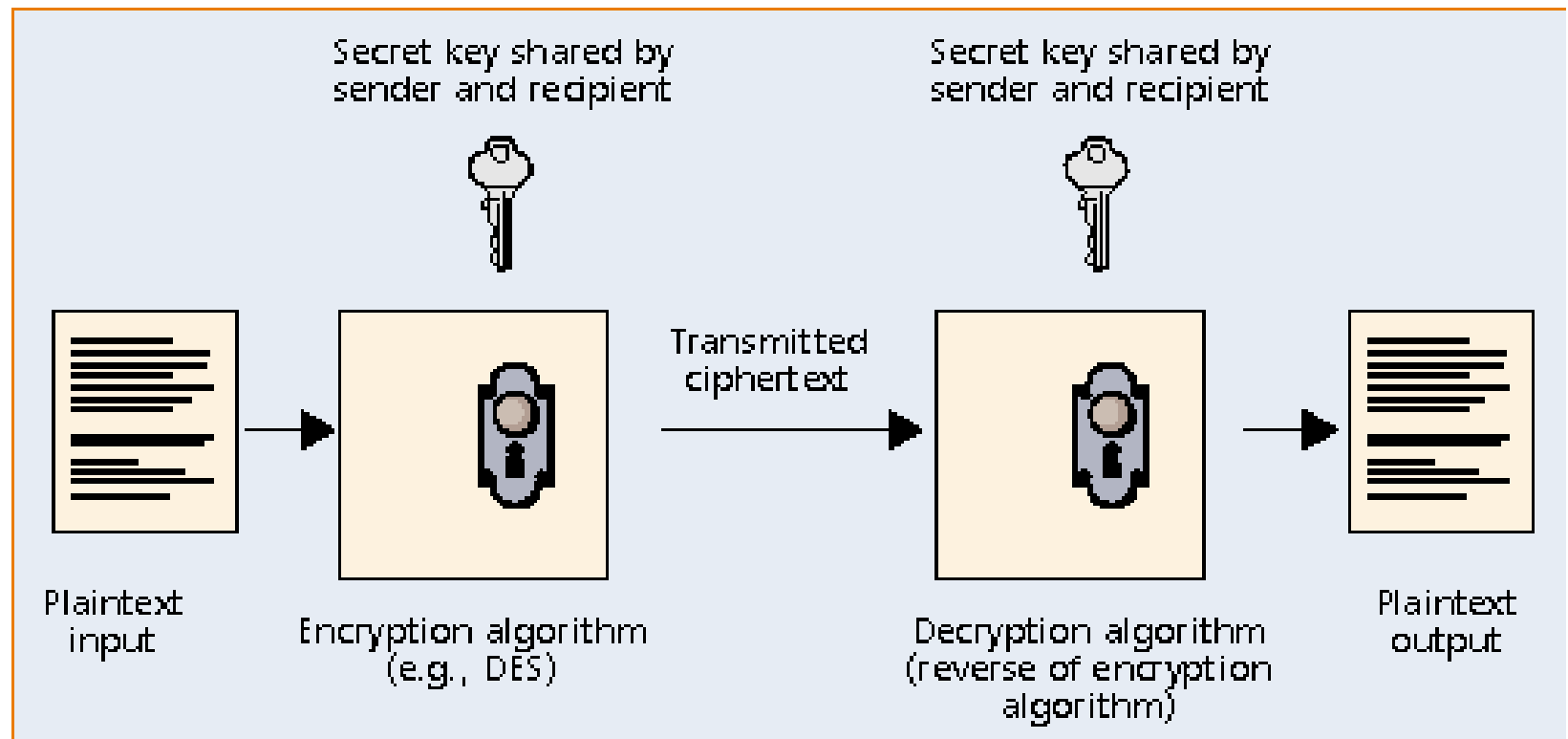


Mecanismos de Segurança Criptográfica - Autenticação



ADD THE MESSAGE AUTHENTICATION CODE (MAC) TO THE DATA
AND SEND THE RESULT

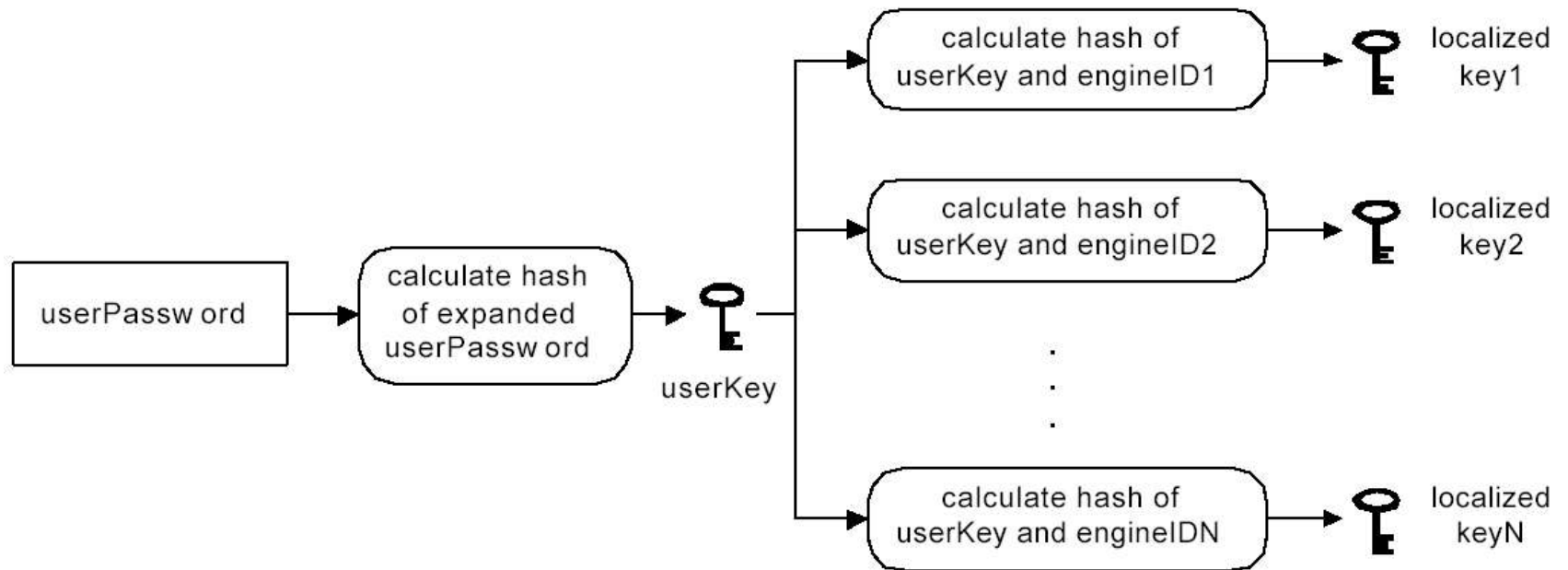
Mecanismos de Segurança Criptográfica – Encriptação DES



Chaves Criptográfica no Modelo de Segurança USM

- Cada usuário possui chaves únicas para ele
- Uma chave para cada serviço de segurança (autenticação e privacidade)
- As chaves criptográficas passam por um processo de localização gerando chaves específicas para cada *engine SNMP*
- Este processo garante que a chave será armazenada de formas diferentes em cada sistema agente. Se uma for comprometida, somente o agente em questão é afetado
- A chave não-localizada não é armazenada em nenhum lugar
- Isto garante que um usuário somente verá um único segredo e os vários *engines* terão segredos independentes

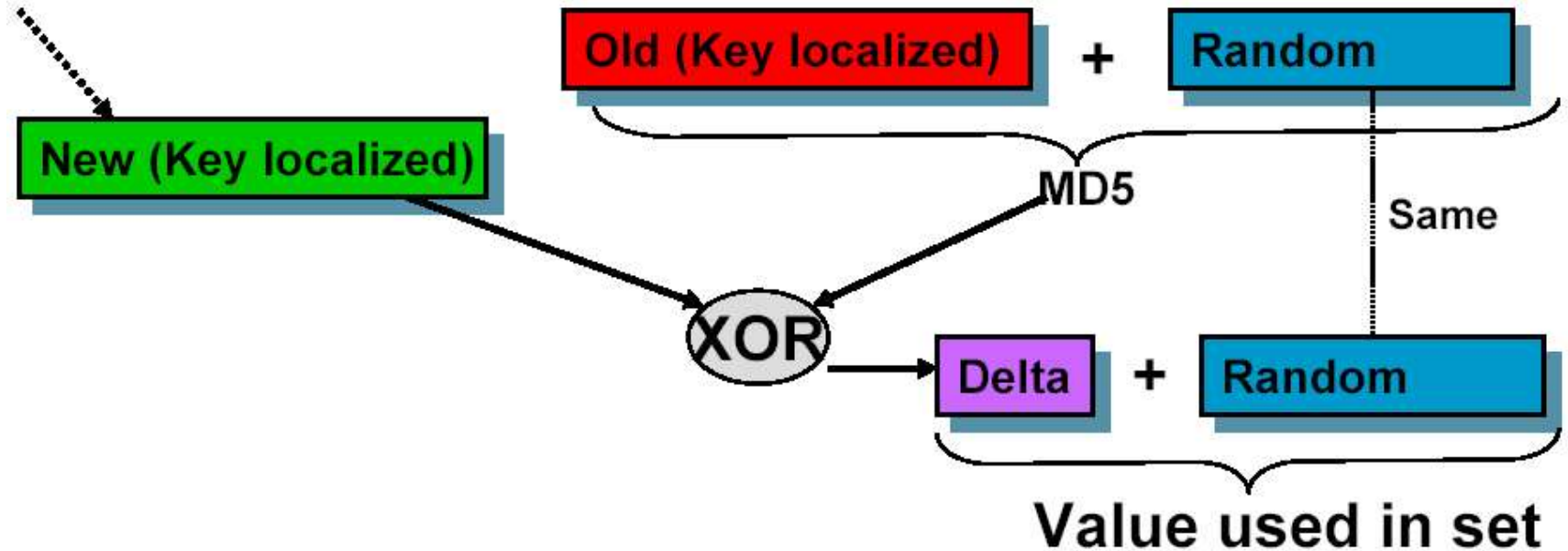
Processo de Criação das Chaves Localizadas



Processo de Atualização das Chaves

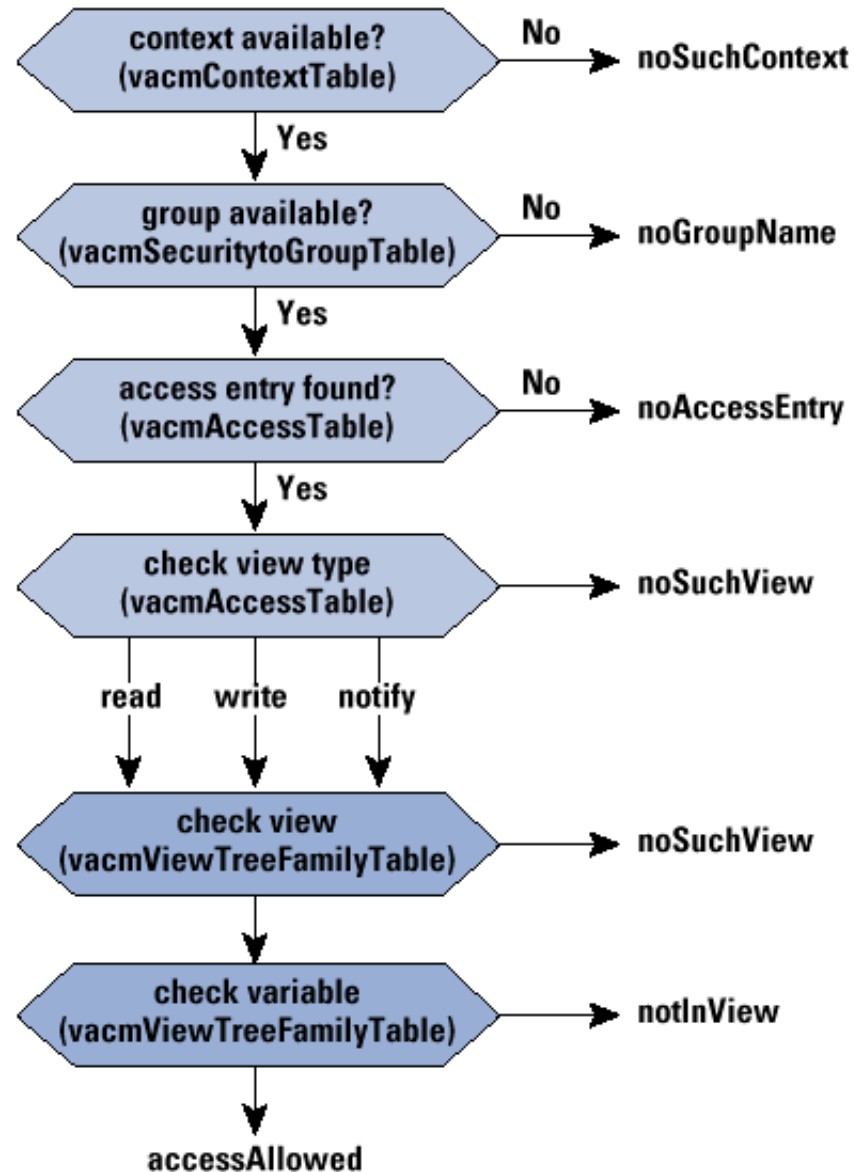
- É implementado um mecanismo de troca de chaves onde é impossível determinar a chave antiga a partir da nova
- Pode já existir no *engine* um material criptográfico inicial pré-configurado (“template”) para servir de base
- A convenção de texto *KeyChange* descreve o procedimento:
 1. Gera-se um valor randômico
 2. Computa-se um valor temporário usando o valor randômico e a chave atual sendo mudada
 3. XOR com a senha desejada, gerando o “delta”
 4. Enviar o valor randômico e o delta para o *engine* a mudar de senha
 5. O *engine* destino refaz a computação e chega no mesmo segredo

Processo de Atualização das Chaves



$(a \text{ XOR } b = c) \text{ implies } (a \text{ XOR } c = b)$

Controle de Acesso



Conclusões

- Trata-se de uma nova tentativa de oferecer segurança e organização, evitando os erros cometidos com SNMPv2p
- Apresenta uma arquitetura modular permitindo evoluções em todos os aspectos - documentação, módulos de processamento de versões, modelos de segurança e até nos algoritmos criptográficos propostos no modelo USM
- Reconhece e trata mensagens SNMP de quaisquer versões
- Utiliza o mesmo PDU SNMPv2
- Já é implementado em vários dispositivos/sistemas