UNIVERSIDADE FEDERAL
DE SANTA CATARINA

ine
CTC • UFSC
Informática e Estatística

INE 5680
Segurança da Informação e de Redes

Autenticação

Profa: Carla Merkle Westphall
carlamw@inf.ufsc.br

Autenticação

RFC 2828 define autenticação do usuário como:

“Processo de verificação de uma identidade reivindicada por ou para uma entidade do sistema.”

Autenticação

- **Primeira linha de defesa**
- **Base do controle de acesso e da contabilização**
- **Fornece garantia de identidade – forma de ganhar a confiança de que pessoas ou coisas são quem ou o que dizem ser**
- **Evita a ameaça do mascaramento (*masquerade*) que pode habilitar as seguintes ameaças:**
  - Negação de serviço
  - Revelação não autorizada de informações
  - Uso ilegítimo
  - Violação de integridade
- ***Principal* – proprietário da identidade legítima**
- **Existem técnicas não criptográficas e criptográficas**

Conceitos Gerais

- **Princípios que guiam a autenticação**
  - Demonstrar ter conhecimento de algo (senha)
  - Demonstrar ter alguma coisa (chave, cartão físico, passaporte)
  - Exibir alguma característica imutável (impressão digital)
  - Apresentar uma evidência de que o requerente está em algum lugar particular (possivelmente em um momento em particular)
  - O verificador aceita que o solicitante já tenha estabelecido uma autenticação
- **Os princípios acima normalmente são combinados para concretizar a autenticação**

Mecanismos de Autenticação

- **Two-Party Authentication**
  - One-way – somente o cliente é autenticado
  - Two-Way – autenticação mútua entre cliente e servidor
- **Two-Party**
  - Autenticação por senhas
    - One-time passwords
  - Challenge-response
  - Smartcard
  - Biometria
- **Trusted Third-Party (TTP) - Autenticação usando servidores de autenticação**

Autenticação baseada em Senhas

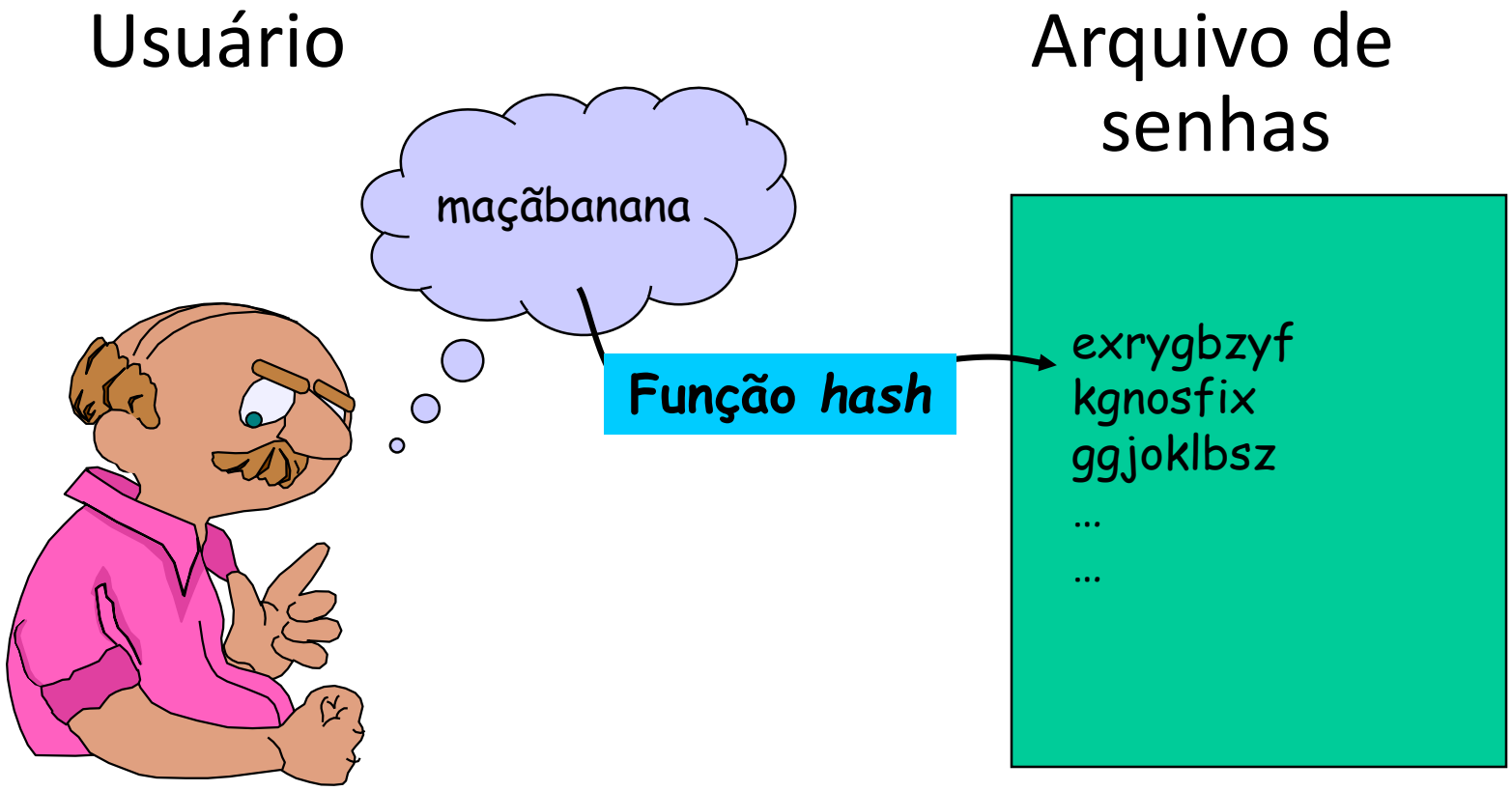
Idéia básica

- Usuários tem uma senha secreta
- O sistema verifica a senha para autenticar o usuário

Vulnerabilidades

- Divulgação externa (*external disclosure*)
- Adivinhação de senhas (*password guessing*)
- Grampo de linha (*line eavesdropping*)
- *Replay*

Esquema básico e fraco de Senhas



Ataque do dicionário

- **Um arquivo de senhas geradas com funções one-way hash é vulnerável**

- **Seqüência de ações de um "ataque do dicionário"**

1) Compilação de uma lista com 1.000.000 de senhas mais comuns
2) Calcular a função one-way em cada uma delas e armazenar o resultado
3) Se cada senha tem 8 bytes, o arquivo de resultado não terá mais de 8 megabytes e pode ser armazenado facilmente
4) Rouba-se um arquivo de senhas. Compara-se o arquivo com o arquivo gerado no passo 3 e verifica-se quais entradas "combinam"

Inserção do Salt (Pitada de Sal ?)

- Protege contra ataques de dicionário mais gerais
- Salt – é uma string aleatória que é concatenada com as senhas antes do cálculo da função one-way
- Ambos os valores de salt e o resultado da aplicação da função one-way são armazenadas no host
- Se o número de valores de salt é grande o suficiente, isso praticamente elimina o ataque de dicionário contra as senhas comumente usadas, porque teria que ser gerado o hash one-way para cada valor de salt possível

Uso de Salt

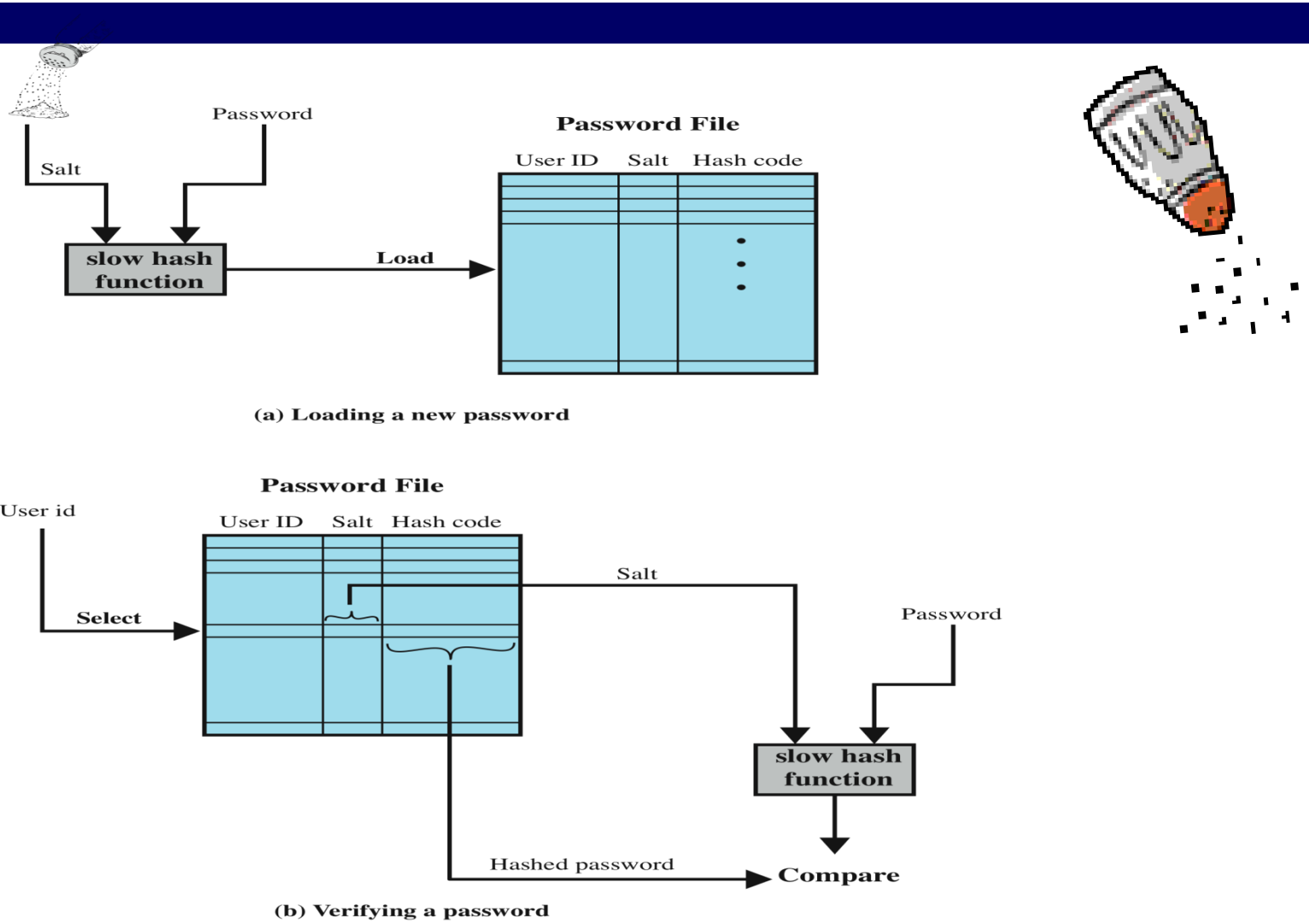
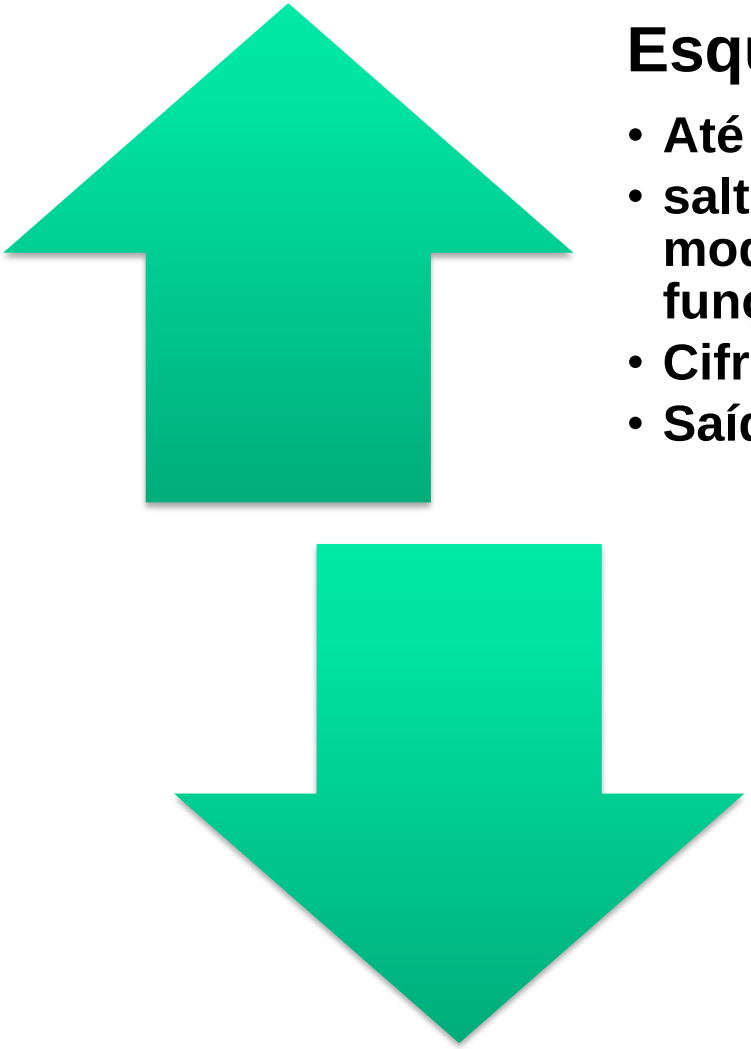


Figure 3.1 UNIX Password Scheme

Ataque “rainbow tables”

- Pré-calcular tabelas de valores de hash para todos os salts
- Gerar assim uma tabela gigantesca de valores de hash
- Podem ser comparados os valores dessa tabela diretamente com cada uma das linhas do arquivo de senhas
- Usam-se valores suficientes de salt e tamanhos de hashes suficientes também

Implementação no UNIX



Esquema original

- Até oito caracteres no tamanho
- salt de 12 bits usado para modificar cifragem DES com função hash
- Cifragem feita 25 vezes
- Saída é sequência de 11 caracteres

Agora é considerado inadequado

- ainda existe por questões de compabilidade

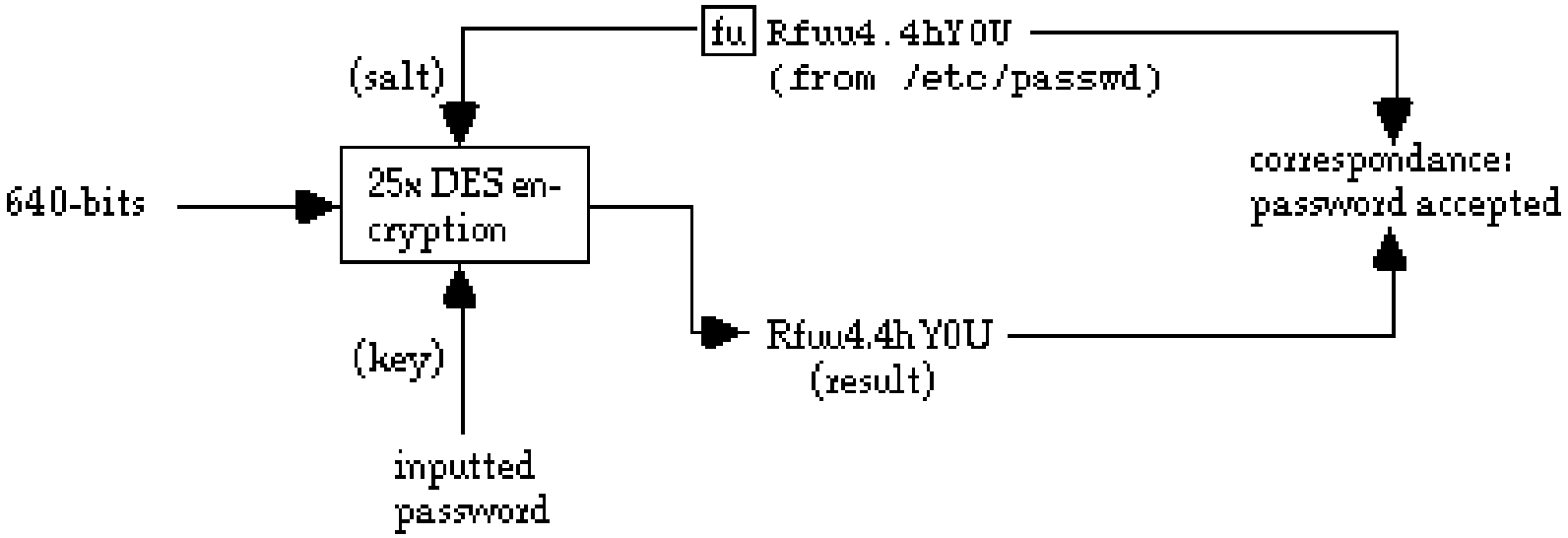
Sistema de senha original do Unix

- Linha de senhas

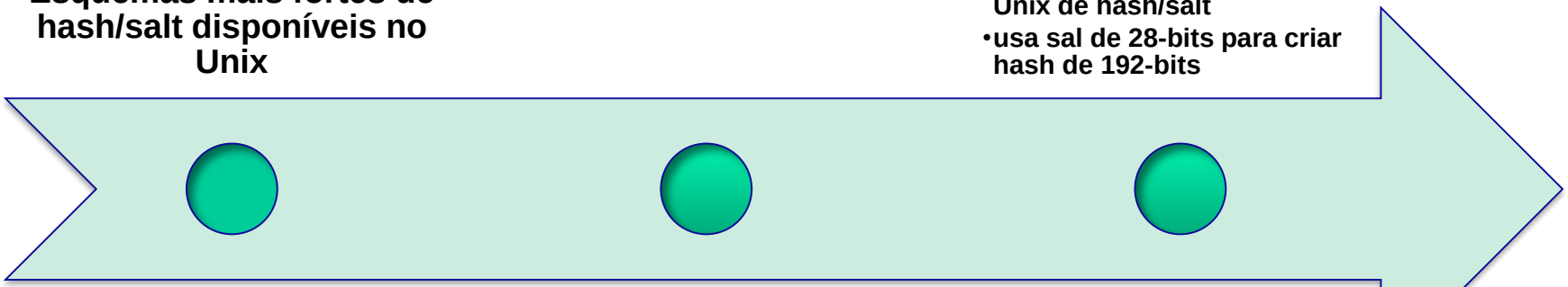
  account:crypted-passwd:uid:gid:user-name:homedir:shell

  walt:fURfuu4.4hY0U:129:129:Belgers:/home/walt:/bin/csh

- Verificando com salt



Implementações melhoradas



Esquemas mais fortes de hash/salt disponíveis no Unix

Função hash baseada em MD5 (já quebrado)

- salt de até 48-bits
- Tamanho ilimitado da senha
- Produz hash de 128 bits
- Usa um loop de 1000 iterações

OpenBSD usa algoritmo de hash chamado bcrypt (baseado no cifrador de bloco Blowfish)

- Mais seguro que esquema do Unix de hash/salt
- usa sal de 28-bits para criar hash de 192-bits

1990- KDF? 2000-PBKDF2? 2009-SCRYPT?
Ver o documento disponível em:
http://www.openwall.com/presentations/Passwords12-The-Future-Of-Hashing/

Autenticação forte

- Definição: Autenticação que requer duas de três características de autenticação (sabe, tem, biometria):
  - Smartcard e pin
  - Biometria e senha
  - Token físico e senha
- Autenticação forte
  - One-time passwords
  - Challenge-Response
  - Baseados em criptografia (Kerberos, X.509,...)

One-time passwords

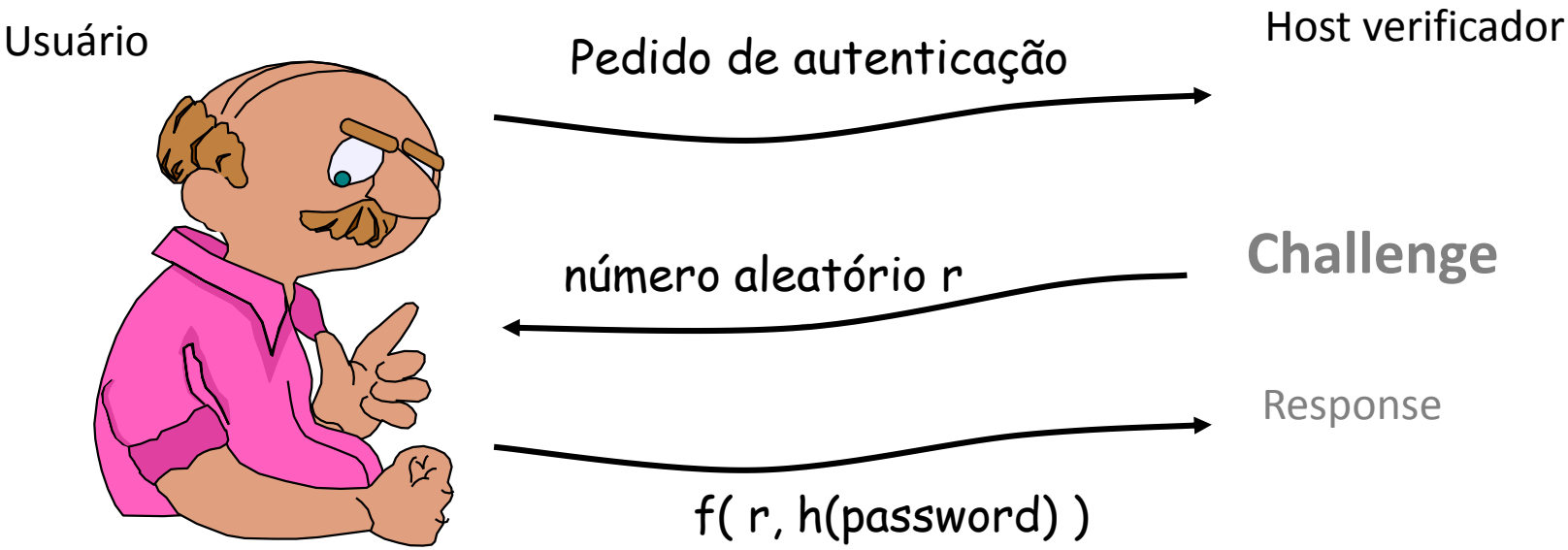
Senhas são usadas uma única vez, garantindo que uma senha diferente seja usada em cada autenticação, a fim de evitar os ataques de *replay*

Esquema do Lamport :

- Não necessita qualquer hardware adicional
- O sistema calcula F(x), F^2(x), ..., F^{100}(x) (isto permite 100 logins antes que a senha mude)
- O sistema armazena (User name, F^{100} (x))
- Usuário fornece F^{99}(x) na primeira vez
- Se o login está correto, o sistema troca F^{100}(x) por F^{99}(x)
- O usuário fornece F^{98}(x) na próxima vez e assim por diante
- Usuário calcula F^n (x) usando: calculadora, estação confiável ...
- Na implementação da Bellcore desse sistema - **S/Key** -, o usuário calcula a seqüência em uma máquina segura, codifica esta seqüência como uma seqüência de palavras pequenas e imprime esta lista

Challenge-response (desafio-resposta)

- Usuário envia identidade
- Host responde com um número aletório r (o desafio!)
- Usuário calcula f(r, h(password)) e envia de volta
- Host compara o valor do usuário com seu próprio valor calculado e se o valor for igual, o usuário está autenticado



NTLM - Windows Challenge/Response

- (autenticação interativa) usuário fornece nome de domínio, nome de usuário e senha. O computador cliente calcula um hash criptográfico da senha e descarta a senha real
- O cliente envia o nome do usuário ao servidor (em texto plano)
- O servidor gera um número aleatório de 16 bytes, chamado challenge (ou nonce) e envia para o cliente
- O cliente cifra o challenge com o hash da senha do usuário e retorna o resultado para o servidor. Isso é chamado de response
- O servidor envia para o controlador de domínio: nome do usuário, challenge enviado para o cliente, response recebido do cliente
- O controlador de domínio usa o nome do usuário para recuperar o hash da senha do BD SAM (Security Account Manager). Usa esse hash para cifrar o challenge
- O controlador de domínio compara o challenge que ele cifra com o challenge calculado cifrado pelo cliente. Caso sejam idênticos, a autenticação teve sucesso
- http://msdn.microsoft.com/en-us/library/windows/desktop/aa378749%28v=vs.85%29.aspx

Protocolos de Autenticação

- **Usados para convencer as partes sobre as identidades de ambas e para trocar chaves de sessão**
- **Pode ser unidirecional ou bidirecional (mútua)**
- **Questões importantes**
  - Confidencialidade – para proteger as chaves de sessão
  - Adequação no tempo (timeliness) – para impedir ataques de replay
- **Para impedir ataques de replay**
  - uso de números de sequência
  - timestamps (precisa relógio sincronizado)
  - challenge/response (usando um nonce único)

Protocolos de Autenticação

- **Seções 3.2 e 3.3 livro Bruce Schneier (tem no moodle)**
- **LER no livro do Stallings: p. 275 até p. 280**
- **ISO 9798-2 – slides anteriores**
- **Tecnicas com criptografia simétrica**
  - Centros de distribuição de chaves (KDC)
  - Protocolo Needham-Schroeder
  - Kerberos (implementação do Needham-Schroeder e de KDC)
- **Tecnicas com criptografia assimétrica**
  - Pressupõe que cada uma das partes está de posse da chave pública da outra
  - Protocolo Denning
  - Protocolo Woo e Lam

SmartCards

- Dispositivo portátil com CPU, portas de I/O, alguma memória não volátil que é acessada através da sua CPU
- Ele é *smart* porque é ativo, pode receber dados, processar e tomar uma decisão
- Alguns usam dados biométricos do usuário ao invés de um PIN
- O primeiro smartcard foi criado por um pesquisado da Bull na França em 1974, e inicialmente eram usados por empresas de telecomunicações e bancos franceses

SmartCards

Leitores de SmartCards

Tecnologias

- SIM: Está presente nos dispositivos GSM
- Mifare: É utilizada pelos cartões de transporte e crachás de acesso
- EMV: (Europay, MasterCard, Visa) É o padrão dos cartões Mastercard e Visa
- ID Card: e-CNPJ, e-CPF

Biométricos

- Impressão digital
- Exame de retina
- Padrão de voz
- Assinatura
- Estilo de escrita

Métodos de Autenticação em PPP

Two-Party Athentication

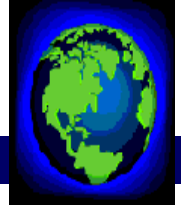
- Password Authentication Protocol (PAP)
- Challenge Handshake Authentication Protocol (CHAP)
- Extensible Authentication Protocol (EAP)
- TACACS
- RADIUS
- Session Key (S/Key)

Radius e TACACS: TCC Luis Cordeiro

Resultado da Autenticação

- Credenciais - Uma credencial designa o principal no sistema e contém os atributos de privilégio e identidade necessários para que o principal possa acessar determinados objetos e serviços no sistema durante uma sessão
- Conjunto de credenciais: identidade (para acesso, auditoria, não-repudiação), papéis (roles), afiliações, níveis de habilitação (clearance)
- Credenciais são usadas por sistemas de autenticação e autorização: SPKI, SDSI, PolicyMaker, Keynote, Shibboleth

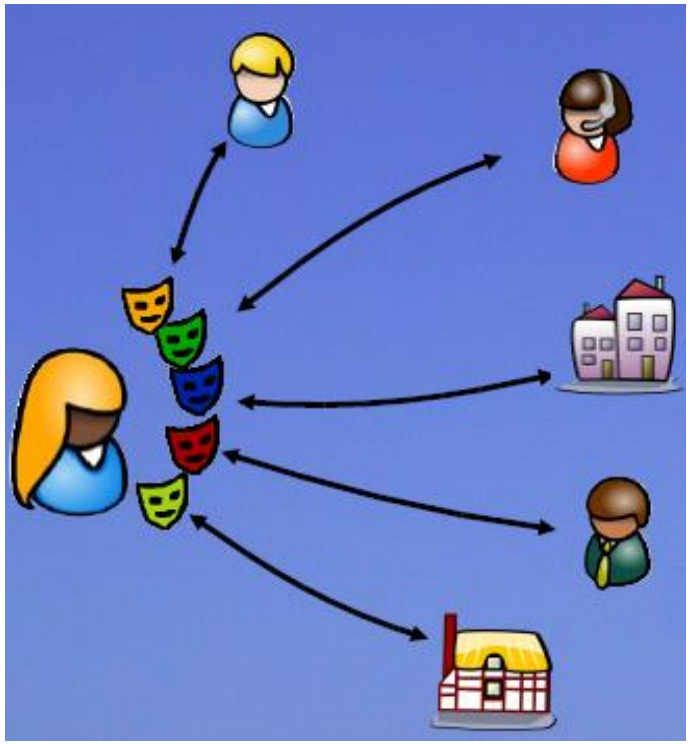
Em sistemas distribuídos...



Múltiplos domínios, múltiplos provedores de serviço...

Uso de sistemas de gerenciamento de identidades!

- Shibboleth
- OpenID Connect
- Oauth
- Kerberos

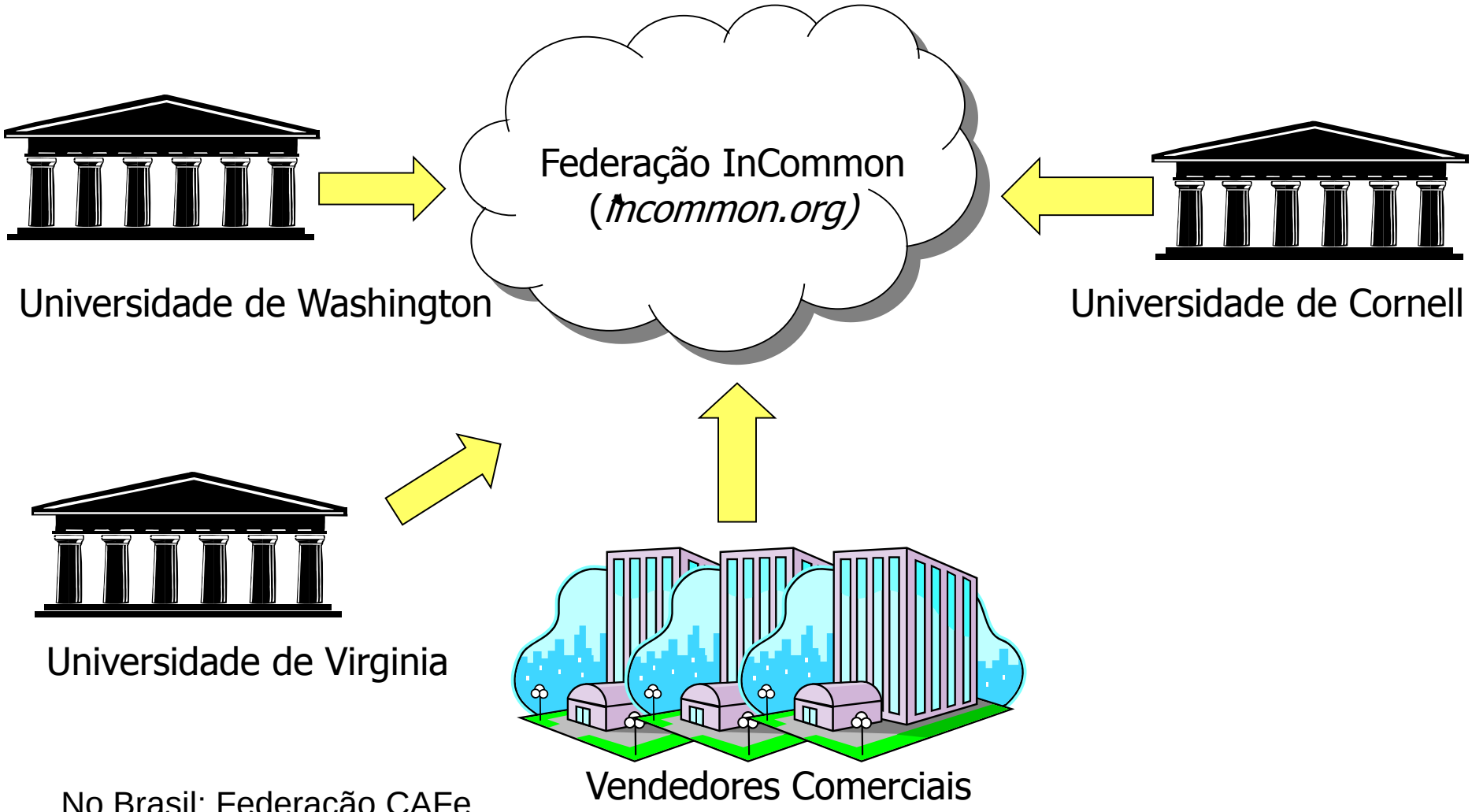


Gerenciamento de Identidades

- Controlar informações de identidade
- Serviços de diretórios
- Federações de organizações
- Autenticação única
- Privacidade dos participantes
- Autorização

http://shibboleth.internet2.edu/

Federação



No Brasil: Federação CAFe

Autenticação

portal.br

Serviço

Origem

Destino

ss77
zx6R-636rs250

zx6R-636rs250

zx6R-636rs250

Posso obter os atributos de zx6R-636rs250

- usuario@portal.br
- t913jtp11@portal.br
- urn:ietf:rfc:1510

Assinado por Origem

Diretório

Attacks	Authenticators	Examples	Typical defenses
Client attack	Password	Guessing, exhaustive search	Large entropy; limited attempts
	Token	Exhaustive search	Large entropy; limited attempts, theft of object requires presence
	Biometric	False match	Large entropy; limited attempts
Host attack	Password	Plaintext theft, dictionary/exhaustive search	Hashing; large entropy; protection of password database
	Token	Passcode theft	Same as password; 1-time passcode
	Biometric	Template theft	Capture device authentication; challenge response
Eavesdropping, theft, and copying	Password	"Shoulder surfing"	User diligence to keep secret; administrator diligence to quickly revoke compromised passwords; multifactor authentication
	Token	Theft, counterfeiting hardware	Multifactor authentication; tamper resistant/evident token
	Biometric	Copying (spoofing) biometric	Copy detection at capture device and capture device authentication
Replay	Password	Replay stolen password response	Challenge-response protocol
	Token	Replay stolen passcode response	Challenge-response protocol; 1-time passcode
	Biometric	Replay stolen biometric template response	Copy detection at capture device and capture device authentication via challenge-response protocol
Trojan horse	Password, token, biometric	Installation of rogue client or capture device	Authentication of client or capture device within trusted security perimeter
Denial of service	Password, token, biometric	Lockout by multiple failed authentications	Multifactor with token