Projeto Final do curso IPv6 a distância

Prova de conceito de controle de acesso na rede física através de autenticação 802.1X numa rede baseada em IPv6

Luciano Ramos
Analista de TI

Jorge Hosni Pereira De Pereira Junior
Analista de TI

Porto Alegre, 26 de agosto de 2013.

SUMÁRIO

1 INTRODUÇÃO 4

2 802.1X - Port Based Network Access Control 6

2.1 CONTROLE DE USUÁRIOS/COMPUTADORES EM REDES 6

2.1.1 Necessidade 6

2.1.2 Visão geral 6

2.1.3 Definições 7

2.1.3.1 *Authenticator Server* 7

2.1.3.2 *Supplicant* 7

2.1.3.3 *Authenticator Switch* ou *Authenticator* 7

2.1.3.4 NAP - *Network Access Port* 7

2.1.3.5 PAE - *Port Access Entity* 8

2.1.3.6 EAP – *Extensible Authentication Protocol* 8

2.1.3.7 *System* 8

2.1.3.8 *PEAP-EAP-TLS* 8

2.1.3.9 *Processo de autenticação do PEAP* 8

2.1.3.10 *PEAP com EAP-TLS* 8

2.1.3.11 *PAP* 9

2.1.3.12 CHAP 9

2.1.3.13 MS-CHAP v1 9

2.1.3.14 MS-CHAP v2 Vantagens 10

2.1.4 Princípio de Operação do protocolo 802.1X 11

2.1.5 Sistema de troca de mensagens 14

2.1.6 Início de uma conexão autenticada 14

3 RADIUS - Remote Authentication Dial In User Service 20

3.1.1 *NPS(Network Policy Server)* 20

4 INFRAESTRUTURA DE CHAVES PÚBLICAS (PKI) 21

Conceito: Uso de conceitos e mecanismos de criptografia de chaves públicas para construção de uma infraestrutura de segurança. 21

4.1.1 Objetivos de uma infraestrutura de segurança: 21

4.1.2 Requisitos da infraestrutura de segurança: 21

4.1.3 Transparência para os usuários: 21

4.1.4 Garantir forma de segurança unificada 21

4.1.5 Componentes da PKI 22

4.1.6 Autoridade Certificadora 22

4.1.7 Certificado 22

4.1.8 Dados de um certificado: 22

4.1.9 Repositório de Certificados 22
4.1.10 Mecanismo para revogar certificados 23
4.1.11 Mecanismo de recuperação e backup de chaves 23
4.1.12 Mecanismo de revalidação de chaves 23
4.1.13 Histórico de chaves 23
4.1.14 Mecanismo para verificação de não-repúdio 23
4.1.15 Marcação de tempo 23
4.1.16 Software Cliente 24
4.1.17 Garantia de autenticidade (para todos os casos) 24
4.1.18 Garantia de autenticidade (Localidade do usuário) 24
4.1.19 Garantia de autenticidade (mecanismos de identificação) 24
4.1.20 Garantia de autenticidade (em uma PKI) 24
4.1.21 Garantia de autenticidade (em uma PKI) 25
4.1.22 Garantia de autenticidade 25
4.1.23 Garantia de integridade 25
4.1.24 Garantia de confidencialidade 25
4.1.25 Considerações 25
4.1.26 Uso de comunicação segura com PKI 25
4.1.27 Uso de garantia de registro de tempo com PKI 26
4.1.28 Garantia de não-repúdio 26
4.1.29 Garantia de não-repúdio 26
4.1.30 Considerações sobre Autenticação e Autorização 26

5 VISÃO RESUMIDA DA IMPLEMENTAÇÃO 27
5.1 TIPOS DE AUTENTICAÇÃO E CERTIFICADOS DE COMPUTADOR E SERVIDOR 27
5.2 CONFIGURAÇÃO DE 802.1X NOS CLIENTES 29
5.3 CONFIGURAÇÃO DE 802.1X NOS CLIENTES 30
5.4 DESENHO RESUMIDO DA SOLUÇÃO EM UMA ORGANIZAÇÃO 30
5.5 DESENHO RESUMIDO DA SOLUÇÃO NO LABORATÓRIO 31

6 PROCEDIMENTOS PARA IMPLEMENTAÇÃO NA REDE DE LABORATÓRIO32
6.1 CONFIGURAÇÃO DO SWITCH 32
6.2 QUANDO FOR AUTENTICAÇÃO POR CERTIFICADO ADICIONAR AS LINHAS ABAIXO NAS PORTAS DOS RESPECTIVOS DISPOSITIVOS: 35
6.3 CONFIGURAÇÃO DOS SERVIDORES 35
6.4 CERTIFICADO DO CLIENTE E CONECTIVIDADE 39

REFERÊNCIAS 41

1 INTRODUÇÃO

Com o aumento de capacidade e a sofisticação dos sistemas de TI (Tecnologia de Informação), o seu grau de complexibilidade também se elevou, causando por consequência, o crescimento exponencial do número de vulnerabilidades, as quais são exploradas com grande rapidez. A esse contexto, acrescenta-se o desenvolvimento de ferramentas e metodologias de ataque, que resultam no cenário atual, em que ataques são propagados em escala mundial, num curto espaço de tempo. Somado a estes fatores, temos o desenvolvimento da nova versão do protocolo IP (*Internet Protocol*), o IPv6 (*Internet Protocol* versão 6), o qual possui grande flexibilidade quanto ao número de endereços, mas como seu antecessor, possui vulnerabilidades que precisam ser inibidas.

Entre o período de surgimento de uma vulnerabilidade e sua correção, atacantes podem penetrar pelas defesas existentes. As ameaças, nesse período, são chamadas de ameaças de dia zero, pois ainda não foi divulgada ao público uma correção ou uma alternativa para o problema. Nesse sentido, convém estabelecer a diferença existente entre dois tipos de solução, a solução reativa, que é aquela em que a reação é tomada após o problema ser detectado, e a pró-ativa, em que o problema é tratado antes mesmo de ocorrer, técnicas como o NDPMon (*Neighbor Discovery Protocol Monitor*) monitoram o ambiente e notificam o administrador, mas não atuam de maneira proativa. Este trabalho objetiva demonstrar a implementação de autenticação 802.1x para prevenção de acesso a rede local, evitando que dispositivos que não estejam autorizados a acessar a rede possam explorar vulnerabilidades.

A fim de explicitar a questão da vulnerabilidade das organizações, é pertinente destacar alguns motivos que tornam suas redes tão facilmente penetráveis, como:

- Possibilidade de falsificação do *Neighbor Discovery*: negação de serviço impedimento de obtenção de endereço IPV6 básico;
- Possibilidade de falsificação do *Router Advertisement*: *Man-in-the-middle* ou negação de serviço por configuração inválida;
- Conteúdo exposto e falta de autenticação: *Man-in-the-middle* ou falsificação de pacotes;
- Varredura de rede;
- Túneis automáticos;
- *Fake relay*, man in the middle;
- Uso de dispositivos pontuais, que não conseguem evitar o DoS *(Denial of Service)*, a propagação de vermes entre servidores, o *Buffer Overflow*, o *Fragmented Attack* e o *Unicode Attack;*
- Emprego de Antivírus sem atualização;
- Presença de *IDS (Intrusion Detector System)* sem assinatura atualizada;
- Demora na disponibilização das correções de vulnerabilidades.

Outro problema constatado é que, embora as correções para uma determinada vulnerabilidade estejam disponíveis, a localização e o isolamento de sistemas comprometidos consomem muito tempo e requerem muitos recursos, tornando as soluções reativas sem eficiência para esse processo. Além disso, durante o período de restauração do ambiente, a rede e os sistemas ficam instáveis. Como consequência desses fatos, as dificuldades para o acesso às correções necessárias tornam-se maiores e mais variadas.

O processo de segurança é composto por todos os elementos dos sistemas de TI, sejam eles servidores, estações de trabalho, equipamentos de rede, aplicações, normas ou procedimentos. Todos eles funcionam como elos de uma corrente, onde a resistência final é igual à resistência do elo mais fraco.

O objetivo deste projeto é implantar um controle de acesso a dispositivos que acessam a rede, nenhum computador sem autorização deve conseguir acesso à rede, impedindo diversos problemas, sejam vírus, ataques, vazamento de informações, entre outros.

2 802.1X - Port Based Network Access Control

2.1 CONTROLE DE USUÁRIOS/COMPUTADORES EM REDES

A grande maioria das redes locais está instalada em ambientes de fácil acesso, que permite a qualquer pessoa utilizar sua infraestrutura, empregando dispositivos não autorizados, através de pontos de conexão disponíveis, ou seja, pontos em que não há controle. Com o objetivo de possibilitar a autenticação dos dispositivos na rede, o *IEEE* (*Institute of Electrical and Electronics Engineers*) desenvolveu o padrão 802.1X, com o qual é possível obter-se uma melhor administração dos recursos existentes, além de aumentar a produtividade através de novas ferramentas que aperfeiçoam os recursos e a infraestrutura. O padrão 802.1X pode ser utilizado para redes baseadas em IPv4 e IPv6.

2.1.1 Necessidade

O crescimento exponencial da tecnologia da informação, mesmo com o novo protocolo IPv6, proporciona uma postura mais ágil na administração de recursos e no aumento de produtividade. Devido a esses fatores, é necessário o desenvolvimento de ferramentas que aperfeiçoem os recursos existentes e implementem técnicas de auditoria e controle à infraestrutura de TI, tendo como marco inicial para que seja possível a aplicação desses paradigmas o surgimento do padrão *IEEE* 802.1X.

2.1.2 Visão geral

Atualmente, um dos principais meios de comunicação é a Internet e, com isso, redes convencionais, incluindo LAN (*Local Area Network*), WAN (*Wide Area Network*) e MAN (*Metropolitan Area Network*), são alvos dos ataques de *hackers*. Entretanto, o que não é levado em conta é que a Internet não é o único meio de acesso dos invasores à rede interna das organizações, uma vez que esses usuários podem valer-se da própria estrutura tecnológica das empresas para obter acesso aos recursos.

A ocorrência desses aspectos está vinculada ao fato de que a grande maioria das redes locais não possui mecanismos de controle de acesso, facilitando o uso da rede de dados por pessoas ou dispositivos não autorizados. Esses invasores, em grande parte das vezes, não possuem usuário e senha para acessar aos servidores, não tendo meios de alcançar o serviço de diretórios contidos nesses equipamentos. Todavia, quando um invasor tem acesso à rede, pode realizar inúmeros ataques, tais como:

IPv4 e IPv6: varreduras de rede, para determinar sistemas ativos com os respectivos sistemas operacionais e serviços em execução; detecção de vulnerabilidades nos sistemas; determinação de pontos fracos conhecidos; invasão de sistemas ou equipamentos de rede através de conexões diretas a recursos compartilhados, cavalos de tróia, ataques ao *SMB* (*Server Message Block*), ataques ao *IIS* (*Internet Information Server*), exploração de vulnerabilidades nas aplicações e ataque de recusa de serviço; tentativa de iniciação de uma sessão de *telnet* no servidor, *switch, router, firewall*, etc; monitoramento do tráfego de rede com analisadores de protocolo; ataque as estações de trabalho; negação de serviço impedimento de obtenção de endereço IPv6 básico; *Man-in-the-middle* ou negação de serviço por configuração inválida; falsificação de pacotes; varredura de rede; túneis automáticos; Fake relay, man in the middle, entre outras.

As vulnerabilidades citadas no parágrafo anterior estão presentes em redes com cabos *(wired)* e em redes sem fio *(wireless),* permitindo que usuários não autorizados utilizem seus serviços, algumas apenas para a versão IPv4 e praticamente todas para IPv6. Essa possibilidade está presente hoje em diversas organizações e este projeto surge para assegurar que a rede está sendo utilizada apenas por aqueles que possuem autorização, utilizando o padrão *IEEE* 802.1X *(Port Based Network Access Control).*

2.1.3 Definições

2.1.3.1 *Authenticator Server*

Agente responsável por fornecer o serviço de autenticação ao *Switch (Authenticator)*, determinando se as credenciais fornecidas pelo *Supplicant* (usuário) estão corretas e, em seguida, autorizando ou não o acesso aos serviços fornecidos pelo *Switch (Authenticator).* Normalmente, esse agente é chamado de servidor RADIUS *(Remote Authentication Dial In User Service).* Em nosso ambiente é utilizado o servidor NPS (*Network Policy Server*) da Microsoft como Proxy RADIUS.

2.1.3.2 *Supplicant*

Dispositivo da rede que necessita ser autenticado, em nosso caso são os clientes utilizando Windows XP, Windows 7 e Windows 8.

2.1.3.3 *Authenticator Switch* ou *Authenticator*

Equipamento que conecta o usuário à rede, tendo a capacidade de mudar sua configuração após a autenticação do usuário.

2.1.3.4 NAP - *Network Access Port*

Porta de conexão do sistema à rede, ou seja, porta do *Switch* em que o usuário está conectado.

2.1.3.5 **PAE - *Port Access Entity***

Entidade do protocolo associada a uma porta física, que pode suportar funcionalidades do protocolo vinculadas ao *Authenticator*, ao *Supplicant* ou aos dois, estando presente em cada porta de um sistema que participa de um *Port based Access Control* e podendo ser: *Supplicant Role*, responsável por responder a requisições de um *Authenticator* (*Switch*), ou seja, pelo fornecimento das credenciais de um *Supplicant* para o *Authenticator* PAE; *Authenticator Role*, responsável pela comunicação com o *Supplicant* e por submeter a informação recebida de um *Supplicant* à um *Authenticator Server*, para que as credenciais sejam checadas, determinando, assim, o estado da autorização.

2.1.3.6 **EAP – *Extensible Authentication Protocol***

Protocolo genérico que permite que os pontos de acesso à rede suportem múltiplos métodos de autenticação, como *smart cards*, TLS (*Transport Layer Security)*, *Kerberos*, entre outros.

2.1.3.7 ***System***

Dispositivo conectado à rede local através de um ou mais pontos, incluindo estações de trabalho, servidores, switches, roteadores, entre outros.

2.1.3.8 ***PEAP-EAP-TLS***

O PEAP (*Protected Extensible Authentication Protocol*) é um novo membro da família dos protocolos EAP (*Extensible Authentication Protocol*). O PEAP usa o TLS para criar um canal criptografado entre um cliente PEAP de autenticação, como um computador sem fio, e um autenticador PEAP, como um serviço de autenticação da Internet (IAS - I*nternet Authentication Service*) ou servidor Remote Authentication Dial-In User Service (RADIUS). O PEAP não especifica um método de autenticação, mas fornece segurança adicional para outros protocolos de autenticação EAP, como EAP-TLS, que podem operar pelo canal criptografado TLS fornecido pelo PEAP.

2.1.3.9 ***Processo de autenticação do PEAP***

Existem dois estágios no processo de autenticação PEAP entre o autenticador e o cliente PEAP. O primeiro estágio configura um canal seguro entre o cliente PEAP e o servidor de autenticação. O segundo estágio fornece a autenticação EAP entre o autenticador e o cliente EAP.

2.1.3.10 ***PEAP com EAP-TLS***

Os certificados de chave pública fornecem um método de autenticação mais forte do que aquele que usa credenciais com base em senha. O PEAP com EAP-TLS (PEAP-EAP-TLS) usa certificados para autenticação de servidores e certificados ou cartões inteligentes ou para autenticação de usuários e de computadores cliente.

2.1.3.11 *PAP*

O PAP *(Password Authentication Protocol)* é um protocolo de autenticação de texto em formato simples. O nome do usuário e senha são esperados pelo servidor de acesso remoto e são enviados pelo cliente remoto em texto de formato simples. Porém, o protocolo PAP não é um protocolo de autenticação seguro. Um usuário remoto que capture pacotes de um segmento de rede aonde está acontecendo uma conexão autenticada por esse protocolo, vai obter de maneira fácil e rápida o usuário e senha entre essa autenticação. Ele também não oferece nenhuma proteção contra ataques de personificação de cliente ou do servidor de autenticação. PAP é um protocolo de troca de mensagens simples:

- cliente de acesso remoto envia uma mensagem de pedido de autenticação PAP ao servidor de acesso remoto contendo o nome de usuário e senha do cliente em texto de formato simples;
- o servidor de acesso remoto então confere o nome de usuário e senha do cliente e envia de volta uma mensagem PAP *Authenticate-Ack* quando as credenciais do usuário estiverem corretas ou uma mensagem PAP *Authenticate-Nak* quando as credenciais do usuário estiverem incorretas.

2.1.3.12 **CHAP**

O CHAP (*Challenge Handshake Authentication Protocol)* é um protocolo de autenticação de desafio de resposta documentado na RFC 1994. Ele usa o protocolo de criptografia MD5 (*Message Digest 5*) de um só sentido para responder a um desafio de resposta hash emitido pelo servidor de acesso remoto. O protocolo CHAP é usado por vários servidores e clientes Dial-up. O protocolo CHAP é uma melhoria em cima dos protocolos PAP pois a senha, nunca é enviada em cima da primeira mensagem. Ao invés, a senha é usada para criar uma string hash de desafio de um só sentido. O servidor sabe a senha do cliente e duplica a operação para comparar o resultado das respostas do cliente. O CHAP é um protocolo de troca de mensagens que usa três mensagens:

- o servidor de acesso remoto envia uma mensagem de desafio CHAP que contém uma chave de sessão e um string hash de desafio arbitrário;
- o cliente de acesso remoto devolve uma mensagem de resposta CHAP que contém o nome de usuário em texto simples, uma string hash de desafio, a chave de sessão e a senha do cliente usando o algoritmo MD5 de um só sentido;
- o servidor de acesso remoto duplica a string hash e compara com a string hash da resposta CHAP. Se as strings hashes são as mesmas, o servidor manda de volta uma mensagem de sucesso CHAP. Se as strings hashes são diferentes, uma mensagem de fracasso CHAP é enviada.

2.1.3.13 **MS-CHAP v1**

O MS-CHAP v1 (*Challenge Handshake Authentication Protocol version 1*) é um protocolo de autenticação codificado bem parecido com o CHAP. Como no protocolo CHAP o servidor de acesso remoto envia um desafio ao cliente remoto que consiste em uma chave de sessão e um string hash de desafio arbitrário. O cliente remoto tem que devolver o nome de usuário e um string hash de desafio em MD4 (*Message Digest 4*), a chave sessão e a senha em MD4 também. Uma diferença entre o CHAP e MS-CHAP v1 é que, no CHAP, a versão de

texto simples da senha deve estar disponível para validar a resposta de desafio. No MS-CHAP v1, o servidor de acesso remoto exige só a string hash MD4 da senha para validar a resposta de desafio. o servidor duplica o string hash e compara com a string hash MS-CHAP resposta do cliente. Se as strings hashes são as mesmas, o servidor envia de volta uma mensagem de Sucesso MS-CHAP. Se as strings hashes são diferentes, uma mensagem de Fracasso MS-CHAP será enviada. O MS-CHAP v1 trabalha com códigos de erro e um deles é, "senha expirou". MS-CHAP v1 protege contra ataques de retomada usando um string hash de desafio arbitrário por tentativa de autenticação. Ele não provê proteção contra personificação de servidor de acesso remoto. O MS-CHAP v1 é um protocolo de troca de mensagens que usa três mensagens:

- o servidor de acesso remoto envia uma mensagem de desafio MS-CHAP que contém uma chave de sessão e um string hash de desafio arbitrário;
- o cliente remoto devolve uma mensagem de resposta MS-CHAP que contém o nome de usuário em texto de formato simples e um string hash de desafio, a chave de sessão e a string da senha em formato MD4 de um só sentido;

2.1.3.14 MS-CHAP v2 Vantagens

- Autenticação de mão dupla. Ela verifica a identidade de ambos os lados da conexão. O cliente de acesso remoto autentica contra o servidor e esse, autentica contra o cliente remoto. Autenticação de mão dupla, também é conhecida como autenticação mútua, pois ela assegura que o cliente remoto está discando para um servidor de acesso remoto que tem acesso à senha do usuário. Autenticação mútua provê proteção contra personificação de servidor remoto;
- Chaves de criptografias separadas são geradas tanto para a transmissão como para recepção dos dados;
- As chaves de criptografia estão baseado na senha do usuário e no string hash de desafio arbitrário. A cada tempo que o usuário se conecta com a mesma senha, uma chave de criptografia diferente é usada

O MS-CHAP v2 é um protocolo de troca de mensagens que usa três mensagens:

- o servidor de acesso remoto envia uma mensagem MS-CHAP v2 de desafio para o cliente de acesso remoto que consiste em uma chave de sessão e um string hash de desafio arbitrário;
- o cliente remoto envia uma mensagem de resposta MS-CHAP v2 contendo: o nome de usuário, uma string hash de desafio arbitrário que ela escutou, uma string hash de desafio que ela recebeu, a chave de sessão e uma string codificada em Message Digest 4 (MD4) da senha do usuário;
- o servidor confere a mensagem resposta MS-CHAP v2 do cliente e envia de volta uma outra mensagem de resposta MS-CHAP v2 contendo: uma indicação do sucesso ou fracasso da tentativa de conexão, uma resposta autenticada baseada na string de desafio enviado, a string de desafio escutada, a resposta codificada do cliente e a senha do usuário.

2.1.4 Princípio de Operação do protocolo 802.1X

Em uma rede com o protocolo 802.1X, implementado cada vez que um usuário *(Supplicant)* quiser acessar aplicações, é necessário, primeiramente, autenticar-se no agente *Authenticator (Switch)* em que está conectado. O *Authenticator Switch* mantém um estado da porta para cada *Supplicant* que reconhecer, ou seja, para cada usuário que se conecta ao equipamento. No caso da autenticação do usuário resultar em uma conexão autorizada, o *Supplicant* pode enviar e receber dados para a rede através do *NAS (Network Access Server),* que é o *Switch* de acesso. Porém, caso o *Supplicant* não tenha sua conexão permitida, o estado da porta permanece não autorizado e o *Supplicant* não pode enviar nem receber dados de suas aplicações na rede. Existem dois meios pelos quais os pacotes vindos de um *Supplicant* podem trafegar, são as chamadas portas de acesso controladas e não controladas.

A figura 2, apresentada na seqüência, ilustra a operação do padrão *IEEE* 802.1X (*Port Based Network Access Control*). Em cada interface física do *Autenticator Switch* são criados dois meios distintos de acesso; as portas controladas e as não controladas. A porta controlada possui dois estados possíveis de autorização, a autorizada, na qual é permitido o acesso aos recursos da rede, e a não autorizada, em que o acesso aos recursos é bloqueado. Já a porta não controlada permite a comunicação sem considerar o estado de autorização. Os pacotes recebidos em uma porta física são disponibilizados para ambas as portas, controladas e não controladas.

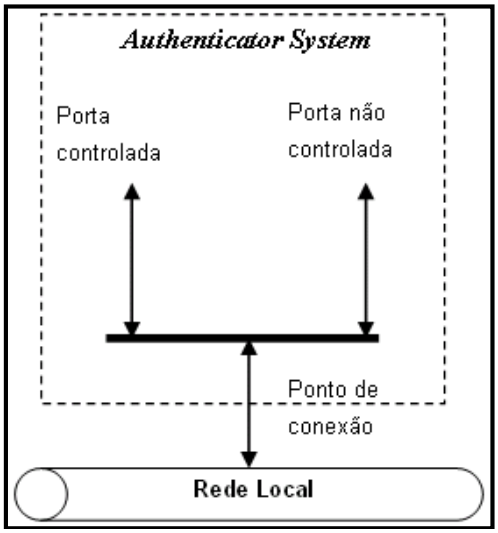


Figura 2 – Porta controlada e não controlada

A figura 3 exibe o estado de uma porta controlada quando o sistema gerenciador faz uma chamada sobre o seu estado *(AuthControlledPortStatus).* O *Switch* pode alterar o estado da porta para conectado ou não conectado, permitindo ou não o fluxo de pacotes, de acordo com sua necessidade. Isso é possível devido a mensagens do tipo *OperControlledDirections*. O parâmetro *AuthControlledPortStatus*, associado à porta controlada do *Authenticator System* 1, mostra que ela está no estado de não autorizada. Já no *Authenticator System 2*, a porta controlada encontra-se autorizada, fazendo com que apareça no estado conectada.

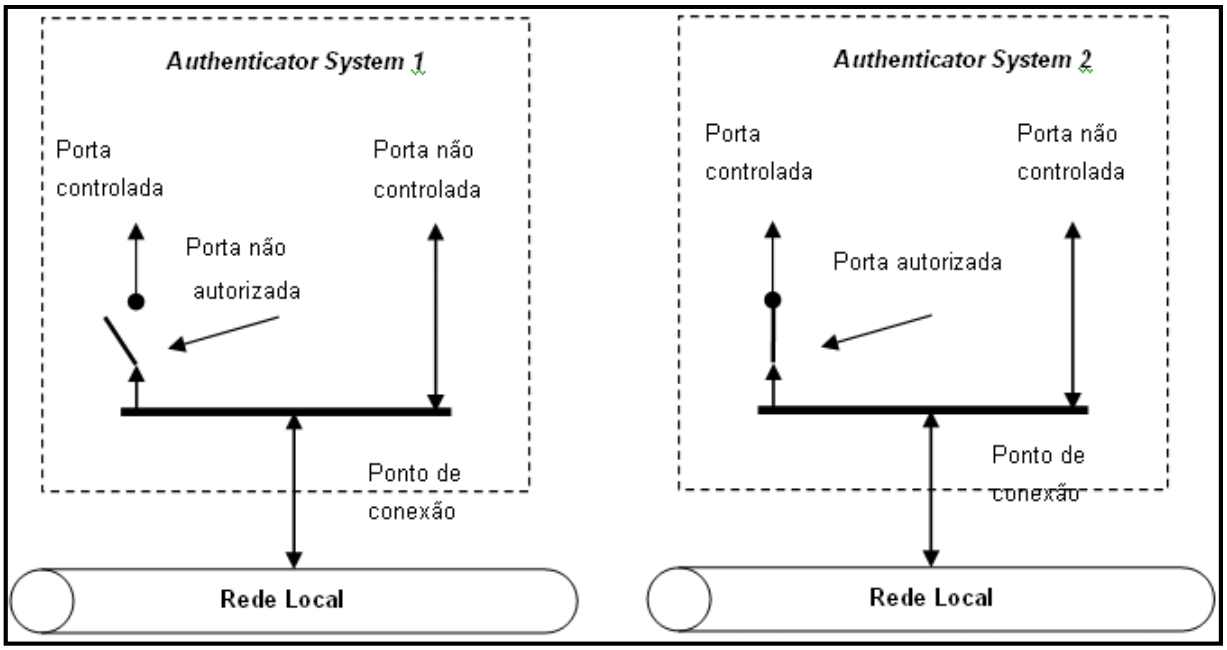


Figura 3 – Mudança de estado em portas controladas

O uso de mensagens *AuthControlledPortStatus* e *AuthControlledPortControl*, em conjunto com o parâmetro associado, permite o controle administrativo sobre a porta. Esse parâmetro pode ter valores *ForceUnauthorized*, *Auto* e *ForceAuthorized*, sendo o segundo considerado como valor padrão. Os relacionamentos entre as mensagens e os parâmetros são de três tipos. No primeiro, uma mensagem *AuthControlledPortControl* com o parâmetro de *ForceUnauthorized* faz com que o *Authenticator PAE (switch)* ajuste o valor da porta controlada para o estado não autorizada, incondicionalmente, não ocorrendo comunicação nesse cenário. No segundo, uma mensagem *AuthControlledPortControl* com o parâmetro de *ForceAuthorized* faz com que o *Authenticator PAE (switch)* ajuste o valor da porta controlada para o estado autorizada, incondicionalmente, tornando desnecessária a autenticação. Por último, uma mensagem *AuthControlledPortControl* com o parâmetro de *Auto* faz com que a o *Authenticator PAE (switch)* ajuste o valor da porta controlada para o estado de acordo com o resultado da troca de mensagens entre o *Supplicant,* o *Authenticator* e o *Authenticator Server.*

Quando se recebe as mensagens na interface física, elas são disponibilizadas para ambas as portas, controladas e não controladas. Os pacotes são proibidos de atravessar a interface da porta controlada, a menos que ela esteja no estado autorizada. Porém, quando os pacotes têm propósito de autenticação, podem atravessar a porta não controlada, a fim de apresentar credenciais de autenticação para o *Authenticator PAE.*

Qualquer acesso à rede está sujeito ao estado administrativo e operacional da porta *MAC*, se essa porta estiver desabilitada, não haverá nenhuma troca de mensagens entre o *Supplicant* e o *Authenticator*, seja nas portas controladas ou nas não controladas. Conforme está ilustrado na figura 4, o *Authenticator System 1* possui as controladas e não controladas habilitadas para o acesso à rede local e o estado administrativo e operacional *MAC* está operante, permitindo, assim, a troca de mensagens entre *Supplicant* e *Authenticator*. Já o *Authenticator System 2* possui o estado administrativo e operacional *MAC* não habilitado, tornando impossível a troca de mensagens entre *Supplicant* e *Authenticator*. O estado inoperante do *MAC* também faz com que o *Authenticator PAE* ajuste a porta controlada para o estado não autorizada.

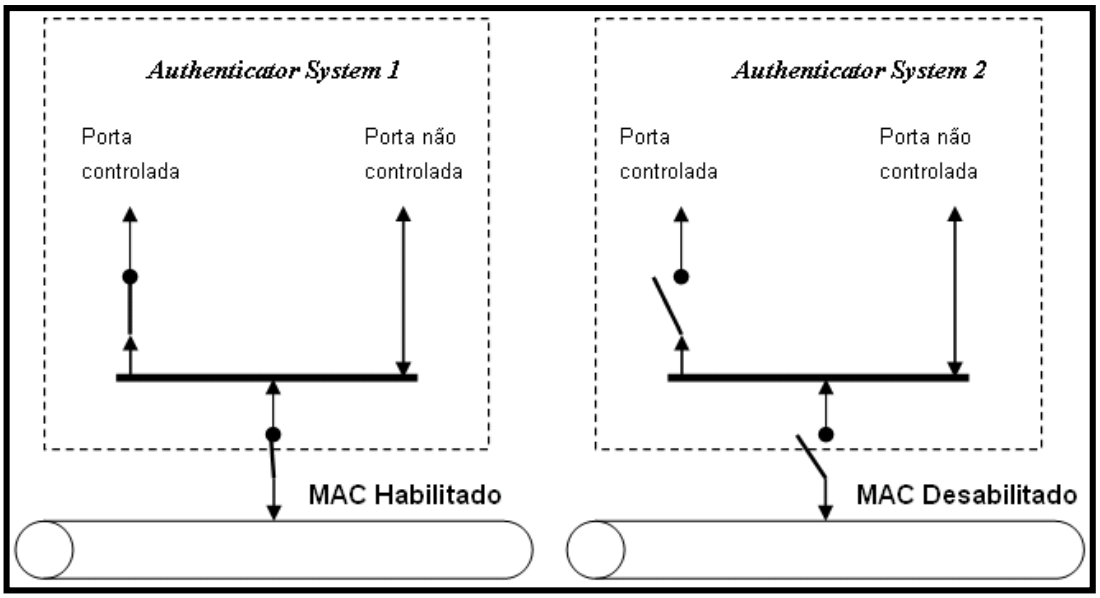


Figura 4 – Efeito de mudança de estado *MAC* nas portas

Em seguida, de acordo com o que representa a figura 5, o *Authenticator PAE* usa as portas não controladas, tendo por propósito a troca de informações com o *Supplicant.* As trocas de informações entre *Authenticator PAE* e *Authenticator Server* (se o servidor não está junto com o *Authenticator PAE*) podem ser realizadas por portas controladas e não controladas.

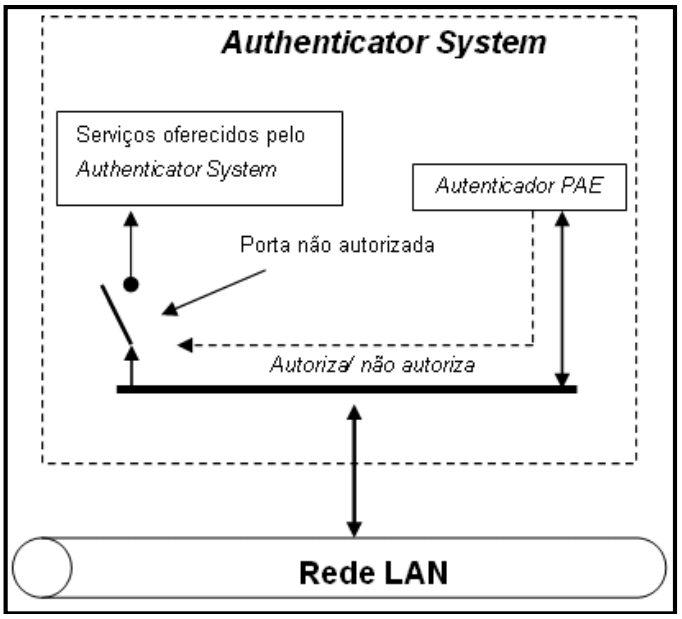


Figura 5 – Uso de portas controladas e não controladas

Depois, a figura 6 exemplifica a relação entre *Supplicant, Authenticator, Authenticator Server* e a troca de informações entre eles. Nessa figura, a porta controlada do *Authenticator* está no estado não autorizada, com relação aos serviços oferecidos pelo *Authenticator System*. Já a troca de mensagens entre o *Supplicant* e o *Authenticator PAE* é feita através de um método utilizado para carregar mensagens *EAP* em um ambiente de rede, chamado *EAPOL* (*Extensible Authentication Protocol Over LANS*). Enquanto isso, a troca de mensagens entre o

Authenticator PAE e o *Authenticator Server* é feita através de protocolos de camada mais elevada, como, por exemplo, *RADIUS*. Caso seja utilizada a porta controlada para comunicação com o *Authenticator Server*, ela deve estar no estado autorizada, pois do contrário, não haverá comunicação para troca de informações.

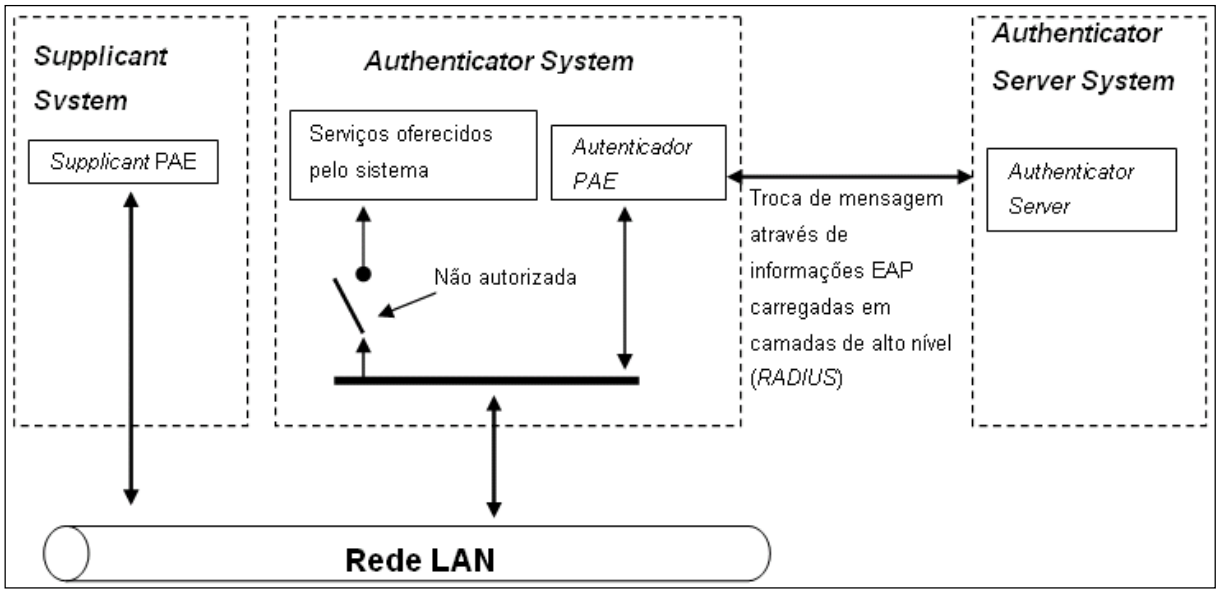


Figura 6 – Troca de mensagens de autenticação

2.1.5 Sistema de troca de mensagens

O processo de autenticação faz uso de mensagens *EAP*, especificadas na *RFC* 2284 (1998), como um meio para comunicar as informações de autenticação entre *Supplicant* e *Authenticator Server*. *EAP* é um protocolo genérico que suporta múltiplos mecanismos de autenticação, tais como *Smart Cards, Kerberos, Public Key Encryption, One Time Passwords*, entre outros.

Conforme descreve o padrão *IEEE* 802.1X, o formato do empacotamento das mensagens *EAP* através de uma rede local é chamado *EAP Over LAN*, ou *EAPOL*. Esse formato é usado para a comunicação entre *Supplicant PAE* e *Authenticator PAE*, em uma rede com o protocolo 802.1X implementado. O *Authenticator PAE* refaz o pacote *EAP*, transmitindo as informações contidas dentro dele em protocolos *RADIUS*, para um *Authenticator Server*.

Mais do que permitir um método de autenticação pré-determinado, o protocolo *EAP* dá condições ao *Authenticator PAE* para que solicite mais informações antes de determinar que tipo de mecanismo de autenticação será utilizado. O *Authenticator PAE* envia uma ou mais requisições para autenticar o *Supplicant PAE* e, nesse pacote de solicitação, existe um tipo de campo que indica o que é requerido. Alguns exemplos de pedidos incluem Identidade, *MD5-challenge, One-Time Passwords,* e *Generic Token Card*, entre outros.

2.1.6 Início de uma conexão autenticada

O processo de autenticação pode ser iniciado tanto pelo *Authenticator PAE* como pelo *Supplicant PAE*. Se a autenticação está habilitada em uma determinada porta, ela é iniciada

pelo *Authenticator PAE* no momento em que ele percebe que ocorreu uma mudança no seu estado operacional, variando de não conectado para conectado. Antes do processo de autenticação ter início, esse dispositivo ajusta a porta controlada para que fique no estado não autorizada.

Para iniciar o processo de autenticação, o *Authenticator PAE* envia uma mensagem *EAP-Request/Identity* (requerimento de identificação) para o *Supplicant*. Visando ao detalhamento desse processo, é implantado um *Sniffer* (ferramenta para diagnóstico de rede), no intuito de analisar os pacotes *EAP* e *RADIUS*, durante o processo de autenticação de um *Supplicant*. A figura 7 mostra que o campo *Code* possui o valor 1, indicando um pacote *EAP* de solicitação (*Request*). O *Identifier* está ajustado para 1 e, assim, é possível fazer a associação entre solicitações e respostas, que devem ter o mesmo valor. O campo *Type* está ajustado para 1, o que indica solicitação de identidade.

```
EAP: ----- EAP Header -----
  EAP:
  EAP: Code = 1 (Request Packet)
  EAP: Identifier = 1
  EAP: Packet length = 1280, DOES NOT EQUAL ACTUAL 42
  EAP:
  EAP: ----EAP data----
  EAP: Type = 1 (identity type)
  EAP: Message = ""
```

Figura 7 – *EAP-Request/Identity*

Em seguida, o *Supplicant PAE* recebe a mensagem *EAP-Request* do *Authenticator PAE* e responde com uma mensagem *EAP-Response.* Conforme a figura 8, o campo *Code* está ajustado para 2, indicando resposta à solicitação anterior, e o *Identifier* está ajustado com o valor 1, associando a resposta atual à solicitação precedente. O *Identity Type* é 1, mas, agora é fornecido o nome do usuário no campo de dados como identidade, sem a senha.

```
EAP: ----- EAP Header -----
  EAP:
  EAP: Code = 2 (Response Packet)
  EAP: Identifier = 1
  EAP: Packet length = 3584, DOES NOT EQUAL ACTUAL 42
  EAP:
  EAP: ----EAP data----
  EAP: Type = 1 (identity type)
  EAP: Message = "empregado"
```

Figura 8 – *EAP-Response* com identidade sem senha

Quando o processo de autenticação é iniciado pelo *Supplicant*, uma mensagem *EAPOL-Start* é enviada para o *Authenticator,* a fim de principiar o processo em que ele, então, emite um *EAP-Request/Identity*. O *Authenticator PAE* é responsável por encaminhar as mensagens *EAP* do *Supplicant* para o *Authenticator Server* e por executar um re-empacotamento dessas mensagens, convertendo aquelas, carregadas como *EAPOL* entre *Supplicant PAE* e *Authenticator PAE* em um formato apropriado para a transmissão ao *Authenticator Server*. Na função de re-encaminhamento, a informação contida dentro das mensagens *EAP* não deve ser modificada, empregando-se o protocolo *RADIUS* para

encapsular informações *EAP* de *Supplicant* na comunicação entre *Authenticator* e *Authenticator Server*.

Ao receber a mensagem *EAP-Response* do *Supplicant*, o *Switch* por sua vez, irá encaminhá-la para um *Authenticator Server* no formato *RADIUS*, convertendo-a em uma mensagem *Access-Request.* Nesse caso, o servidor *RADIUS* estará trabalhando como um identificador e fornecendo um perfil de acesso. A figura 9 ilustra o *Access-Request*.

```
RADIUS: ----- RADIUS HEADER -----
RADIUS:
RADIUS: Code          = 1 (Access-Request)
RADIUS: Identifier    = 26
RADIUS: Length        = 108
RADIUS: Authenticator = CF320000A24D0000161D00002C2E0000
RADIUS:
RADIUS: Attributes follow
RADIUS: Attribute Type     = 80
RADIUS: Attribute Length   = 18
RADIUS: Message-Authenticator = "ò8$ì&ÕD´ßs~Ýz3ùU"
RADIUS:
RADIUS: Attribute Type     = 1
RADIUS: Attribute Length   = 11
RADIUS: User-Name          = "empregado"
```

Figura 9 – *Access-Request*

Em resposta ao *Access-Request*, o *Authenticator Server* envia ao *Authenticator System* um desafio, ou seja, um *Access-Challenge*, como está exposto na figura 10.

```
RADIUS: ----- RADIUS HEADER -----
RADIUS:
RADIUS: Code          = 11 (Access-Challenge)
RADIUS: Identifier    = 26
RADIUS: Length        = 100
RADIUS: Authenticator = FF6CB2BB544582E8863DD27B8F89C768
RADIUS:
RADIUS: Attributes follow
RADIUS: Attribute Type     = 27
RADIUS: Attribute Length   = 6
RADIUS: Session-Timeout = 6
RADIUS:
RADIUS: Attribute Type     = 79
RADIUS: Attribute Length   = 31
RADIUS: EAP-Message = "..."
RADIUS:
RADIUS: Attribute Type     = 24
RADIUS: Attribute Length   = 25
RADIUS: State = "...à"
RADIUS:
RADIUS: Attribute Type     = 80
RADIUS: Attribute Length   = 18
RADIUS: Message-Authenticator = "¨.▌▌$^8«mó.O)±.▌"
```

Figura 10 – *Access-Challenge*

O *NAS*, nesse momento, encaminha uma mensagem *EAP-Request* com o desafio para o dispositivo *Supplicant*. O campo de dados do pacote *EAP* tem ajustada a opção 4, indicando que uma *String* é emitida de um *Authenticator Server* para um *Supplicant*, em uma mensagem *MD5-Challenge Request*. Isso pode ser constatado na figura 11.

```
EAP: ----- EAP Header -----
    EAP:
    EAP: Code = 1 (Request Packet)
    EAP: Identifier = 2
    EAP: Packet length = 7424, DOES NOT EQUAL ACTUAL 42
    EAP:
    EAP: ----EAP data----
    EAP: Type = 4 (md5_challenge type)
    EAP: Value Size = 16
    EAP: Value = "▌ .-áéJ.▌.Ý:ûÛÅ."
    EAP: name = "RADISEC"
```

Figura 11 – *EAP-Request* com desafio

A *Challenge String* com usuário e senha é *hashed*, utilizando o algoritmo *MD5*, sendo que o *hash* resultante retorna em uma mensagem *MD5-Challenge Response*, como ilustra a figura 12.

```
EAP: ----- EAP Header -----
    EAP:
    EAP: Code = 2 (Response Packet)
    EAP: Identifier = 2
    EAP: Packet length = 7936, DOES NOT EQUAL ACTUAL 42
    EAP:
    EAP: ----EAP data----
    EAP: Type = 4 (md5_challenge type)
    EAP: Value Size = 16
    EAP: Value = "È"
    EAP: name = "empregado"
```

Figura 12 – *EAP-Response* ao desafio

A seguir, o *Authenticator* encaminha esse *hash* para o *Authenticator Server*, como exemplifica a figura 13.

```
RADIUS: ----- RADIUS HEADER -----
    RADIUS:
    RADIUS: Code          = 1 (Access-Request)
    RADIUS: Identifier    = 27
    RADIUS: Length        = 150
    RADIUS: Authenticator = E5400000886200001D670000F53F0000
    RADIUS:
    RADIUS: Attributes follow
    RADIUS: Attribute Type     = 80
    RADIUS: Attribute Length   = 18
    RADIUS: Message-Authenticator = "▌<▌8ÄF.D▌Z®xíÅ¤."
    RADIUS:
    RADIUS: Attribute Type     = 1
    RADIUS: Attribute Length   = 11
    RADIUS: User-Name          = "empregado"
```

Figura 13 – *Access Request*

No momento subseqüente, o *RADIUS* determina se o *Supplicant* possui ou não acesso à rede, executando o mesmo processo e comparando o resultado com o retorno da mensagem *Access-request*, a fim de definir se há sucesso ou falha na autenticação. No caso de o usuário ser identificado, é enviado um *RADIUS-Accept* para o *NAS*, conforme se observa na figura 14.

```
RADIUS: ----- RADIUS HEADER -----
    RADIUS:
    RADIUS: Code          = 2 (Access-Accept)
    RADIUS: Identifier    = 27
    RADIUS: Length        = 132
    RADIUS: Authenticator = 52A0ED716199829D021E62B7B6A7E723
    RADIUS:
    RADIUS: Attributes follow
    RADIUS: Attribute Type     = 11
    RADIUS: Attribute Length   = 44
    RADIUS: Filter-Id = "Enterasys:version=1:policy=Empregado Geral"
```

Figura 14 – *Access-Accept*

O *NAS,* por sua vez, encaminha uma mensagem *EAP-Success* para o dispositivo, autorizando o tráfego de dados na respectiva porta do *Supplicant System* (usuário), como expõe a figura 15.

```
EAP: ----- EAP Header -----
    EAP:
    EAP: Code = 3 (Success Packet)
    EAP: Identifier = 2
    EAP: Packet length = 1024, DOES NOT EQUAL ACTUAL 42
    EAP:
```

Figura 15 – *EAP-Success*

Esse processo acontece toda vez que é constatada uma mudança de estado na porta e isso significa que, caso o usuário desconecte seu dispositivo da porta, todas as configurações serão perdidas, sendo necessária a repetição de todo o processo de autenticação. Todas as

mensagens que o *Authenticator PAE* receber do *Authenticator Server* devem ser convertidas para o formato *EAPOL*, a fim de que seja possível transmitir para o *Supplicant PAE*.

3 RADIUS - Remote Authentication Dial In User Service

O *RADIUS* é um protocolo utilizado para carregar informações de autenticação e autorização entre um *NAS (Switch)* e um *Authentication Server (RADIUS).* O *NAS* age como um cliente do *RADIUS,* fornecendo uma interface entre usuários finais, que estão requisitando serviços de autenticação, e um servidor *RADIUS,* que verifica as credenciais dos usuários. Normalmente, esse serviço utiliza a porta *UDP* 1812.

O protocolo *RADIUS* constitui-se de três partes principais (*RFC* 2138, 1997):

- Autenticação: verificação das credenciais do usuário para determinar sua identidade, que pode ser uma etapa simples, como, por exemplo, o fornecimento de usuário e senha, ou complexa, havendo necessidade de informações complementares de *"smart-cards"* ou cartões com senhas transitórias;
- Autorização: processo que estabelece quais os recursos da rede que o usuário poderá acessar, definindo as propriedades da conexão liberadas pelo processo de autenticação, além de permitir a existência de usuários com privilégios distintos;
- Contabilidade: registro de utilização dos recursos de rede, como, por exemplo; tempo de conexão e banda, entre outros.

3.1.1 *NPS(Network Policy Server)*

O *NPS* é uma aplicação, incorporada ao *Microsoft Windows 2008 (substituto do IAS)*, que fornece o serviço de *RADIUS*, no qual é criada uma política de acesso remoto além de outras funcionalidades. *NPS* executa autenticação, autorização e contabilidade de muitos tipos de acesso, também funciona como Proxy Radius. O servidor NPS também funciona como um validador de conformidades, verificando se o cliente esta com todos os pré-requisitos de controle habilitados, como por exemplo, antivírus, atualizações de sistema, etc.

O NPS possui incluído em sua aplicação o NAP ou Network Access Policy o qual é uma tecnologia introduzida no Windows Server 2008 que permite validar a "saúde" dos hosts que conectam em sua rede.

Através do NAP é possível verificar se o host que esta se conectando a rede através do protocolo 802.1X, possui um software de Antivírus instalado e atualizado ou se o serviço de atualizações automáticas esta iniciado. Caso o não esteja dentro dos padrões estipulados com as políticas de acesso a rede, ele é direcionado a servidores de remediação, permitindo corrigir os problemas de segurança encontrado e em seguida tentar um novo acesso a rede.

A vantagem do NAP é que ele é expansível, ou seja, novas validações podem ser adicionadas a ele e impostas aos hosts na rede.

4 INFRAESTRUTURA DE CHAVES PÚBLICAS (PKI)

Conceito: Uso de conceitos e mecanismos de criptografia de chaves públicas para construção de uma infraestrutura de segurança.

4.1.1 Objetivos de uma infraestrutura de segurança:

- Possibilitar uma aplicação ou um tipo de aplicação com acréscimo de segurança aos dados, sistemas ou recursos;
- Possibilitar uma aplicação ou um tipo de aplicação com segurança na interação entre dados, sistemas ou recursos.

4.1.2 Requisitos da infraestrutura de segurança:

- Interface amigável, intuitiva;
- O resultado de seu uso deve ser previsível e útil;
- Modo de funcionamento conhecido pelos recursos e sistemas.

4.1.3 Transparência para os usuários:

- A infraestrutura de segurança deve ser transparente ao usuário final;
- Nenhum tipo de conhecimento técnico deve ser necessário ao usuário;
- Nenhum mecanismo de controle complexo deve ser adicionado na vida do usuário;
- Um usuário deve ser capaz de realizar suas tarefas sem se preocupar em como a segurança é atingida.

4.1.4 Garantir forma de segurança unificada

- Os diversos mecanismos, sistemas e servidores devem compartilhar a mesma tecnologia de segurança para:
    - Troca de dados;
    - Armazenamento de dados;
    - Interação com usuários;

- Processamento de transações;
- Acesso aos serviços.

4.1.5 Componentes da PKI

- Autoridade Certificadora
- Repositório de Certificados
- Mecanismo para revogar certificados
- Mecanismo de recuperação e backup de chaves
- Mecanismo de revalidação de chaves
- Histórico de chaves
- Certificação cruzada (entre PKIs)
- Mecanismo para não-repúdio
- Marcação de tempo (*timestamp*)
- Software Cliente

4.1.6 Autoridade Certificadora

Dois usuários desconhecidos devem ser capazes de trocar dados de forma confiável e segura, tem-se a necessidade de uma entidade/autoridade que ambos "confiem".

A CA é capaz de certificar a chave pública de um usuário assinando digitalmente esta chave, ao receber uma chave certificada pela CA, pode-se confiar nela.

4.1.7 Certificado

Um certificado é a consolidação de uma identificação de usuário e um par de chaves.

4.1.8 Dados de um certificado:

- Nome do usuário ou entidade associada à chave;
- Período de validade do certificado;
- Chave pública;
- Nome e assinatura da CA;
- Número de série.

4.1.9 Repositório de Certificados

Um usuário deve poder obter a chave certificada (certificado) de outro usuário de maneira fácil. Necessidade da criação de um local de armazenamento de certificados.

4.1.10 Mecanismo para revogar certificados

Existem situações onde um certificado deve ser revogado, ex:

- Troca de identidade de usuário;
- Perda de chave privada;
- Desligamento de usuário da infraestrutura de segurança.

4.1.11 Mecanismo de recuperação e backup de chaves

- Perda de mídia (onde estava a chave privada);
- Usuário esqueceu senha de acesso;
- Perda do arquivo da chave privada.

4.1.12 Mecanismo de revalidação de chaves

- O certificado possui um tempo de validade;
- Deve haver um mecanismo capaz de revalidar certificados expirados;
- Um novo certificado é gerado com uma nova data de validade.

4.1.13 Histórico de chaves

- Com a renovação de certificados, acaba-se criando um histórico de todos os certificados e chaves emitidos para um usuário;
- Necessário para que se consiga decifrar dados cifrados com uma chave antiga;
- A escolha da chave correta deve ser transparente para o usuário.

4.1.14 Mecanismo para verificação de não-repúdio

- Usuários de uma PKI não podem romper a relação com sua chave e a CA;
- Um usuário não deve ser capaz de negar a autoria de um dado assinado pela sua chave;
- A PKI deve fornecer subsídios em forma de evidência para evitar o não-repúdio;
- obviamente em caso de roubo de chaves, o administrador terá a decisão final.

4.1.15 Marcação de tempo

- Deve existir algum serviço de marcação de tempo confiável, ao qual os sistemas que utilizam a PKI possam referenciar;
- útil em casos de troca de dados, garantia de não-repúdio e controle de tempo em cifragem e assinatura de documentos (controle de dead-lines.)

4.1.16 Software Cliente

- Os sistemas onde a PKI está inserida devem possuir recursos para o uso das chaves;
- Devem possuir a capacidade de interpretar/utilizar o histórico de chaves, recuperação e atualização de certificados;
- Estes sistemas devem possibilitar:
    - Uso das chaves para acesso aos dados
    - Uso das chaves para cifragem de dados
    - Uso das chaves para assinatura

4.1.17 Garantia de autenticidade (para todos os casos)

- Garantir que um determinado elemento é realmente quem diz ser;
- Evitar falsificação de identidades de sistemas ou usuários;
- Associada à identificação está a liberação de uma chave para decifrar uma determinada informação, permissões de acordo com ACLs, alteração de dados, estabelecer a origem de uma determinada informação.

4.1.18 Garantia de autenticidade (Localidade do usuário)

- Local:
    - Usuário está fisicamente próximo ao ambiente ou dispositivo de identificação;
    - Informação de identificação dentro de limites “mais seguros” .
- Remoto:
    - Informação deve atravessar a rede para validar a identificação.

4.1.19 Garantia de autenticidade (mecanismos de identificação)

- Algo que o usuário possui (como um smartcard, token);
- Algo que o usuário sabe (senha, segredo);
- Algo físico do usuário (biometria);
- Algo que o usuário faz (identificação da letra/estilo de escrita);
- pode-se utilizar mais de um mecanismo simultaneamente.

4.1.20 Garantia de autenticidade (em uma PKI)

- Inicialmente deve-se ter acesso ao par de chaves a chave privada deve estar protegida;
- Dispositivos para armazenamento da chave privada tokens smartcards arquivo no computador (perigoso em caso de roubo, perda ou invasão);
- A liberação do uso do par de chaves DEVE ser protegida por senha de acesso.

4.1.21 Garantia de autenticidade (em uma PKI)

Após a liberação da chave, a autenticação de um usuário utiliza a PKI

Ex: Bob quer se autenticar com Alice:

- Alice envia um texto desafio para Bob
- Bob assina o texto com sua chave privada
- Alice verifica a assinatura com a chave pública de Bob

4.1.22 Garantia de autenticidade

Deve-se garantir que o transporte dos dados seja protegido. Cuidar com ataques de replay. Utilizar serviço de tempo (quando disponível).

4.1.23 Garantia de integridade

Qualquer tentativa de alteração de dados deve ser detectada.

Ex: Para que Alice verifique a integridade dos dados de Bob, este pode assinar um digest da mensagem (HASH), podendo também ser utilizada uma sessão simétrica.

4.1.24 Garantia de confidencialidade

Os dados devem estar protegidos do acesso não autorizado. Isto serve tanto para dados em trânsito, quanto armazenados, de backup ou não.

Ex: Para que os dados enviados por Bob não sejam interceptados de maneira legível, esse deve cifrá-los com a chave pública de Alice, ou utilizando uma chave simétrica trocada de maneira segura. Ex: criptografia mista

4.1.25 Considerações

Criptografia simétrica é mais performática que assimétrica;

Existe a possibilidade de utilização de pares de chaves efêmeras, quando conveniente:

- Alice gera um par de chaves temporário, enquanto Bob utiliza as chaves pré-armazenadas
- Tanto Alice, quanto Bob geram um par de chaves temporário

O algoritmo criptográfico a ser utilizado pode ser negociado entre as partes

4.1.26 Uso de comunicação segura com PKI

- Browser: protocolo TLS (com autenticação mútua entre cliente e servidor)
- E-mail: clientes aceitam PKI para cifrar e assinar mensagens
- VPN: Ipsec (entre outros) com troca de chaves

4.1.27 Uso de garantia de registro de tempo com PKI

- O sistema responsável pelo *timestamp* pode garantir a referência de tempo de uma informação através da garantia de:
- Integridade do *timestamp* (HASH com o arquivo);
- Autenticidade do *timestamp* (assinatura do arquivo c/ *timestamp*).

Ex: deadline de um relatório, trabalho de um aluno, licitação, ..

4.1.28 Garantia de não-repúdio

- Da origem da informação
- Do recebimento de uma informação
- Da criação de uma informação
- Da entrega de uma informação
- Do aceite de uma informação
- Detecção de malícia ou negligência de usuários

4.1.29 Garantia de não-repúdio

- Não pode ser baseada em chaves simétricas (Alice e Bob possuem conhecimento da chave);
- Requer o *timestamp* da ação;
- Mais eficiente se existir a validação da informação;
- Baseia-se também na existência do mecanismo de registro de histórico de chaves;
- É um serviço complexo de ser atingido;
- Roubo de chaves deve ser considerado.

4.1.30 Considerações sobre Autenticação e Autorização

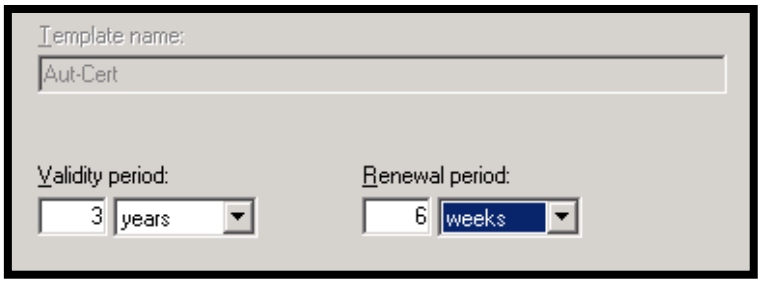
- Autenticação: prova de que um usuário é realmente quem alega ser
- Autorização: o quê este usuário pode fazer (ACL)
- Delegação: passagem de autorizações entre usuários
- Política: regras para todos os aspectos acima

5 VISÃO RESUMIDA DA IMPLEMENTAÇÃO

5.1 TIPOS DE AUTENTICAÇÃO E CERTIFICADOS DE COMPUTADOR E SERVIDOR

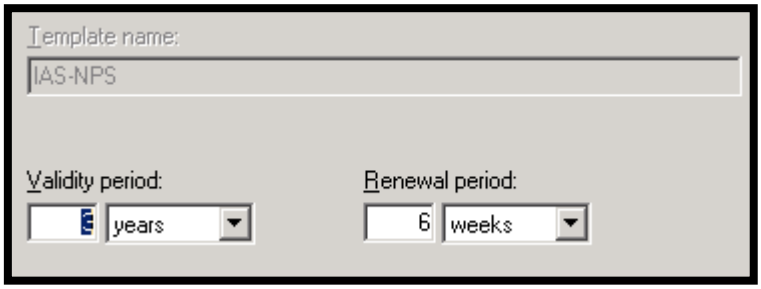
Dois tipos de autenticação são utilizadas nas portas do switch da rede, em computadores com Windows XP, Windows 7 e Windows 8 é utilizado PEAP-EAP-TLS, nos demais dispositivos como por exemplo impressoras, cameras e codins é utilizado controle por MAC Address.

Como pré-requisito foi configurada uma Autoridade Certificadora 2008. Foram configurados templates de certificado para desktops e servidores RADIUS. Certificados de computador foram configurados com um prazo de validade de 3 anos e um período de renovação a cada 6 semanas, essa configuração é ajustável, conforme figura abaixo:



Certificado de Computador

Já para os certificados dos servidores de autenticação também foram configurados com um prazo de validade de 3 anos e um período de renovação a cada 6 semanas, conforme segue abaixo:



Certificado dos Servidores Radius

Foram ajustados os templates para que os certificados de computador e servidor fossem distribuídos automaticamente.

Maiores detalhes sobre instalação, configuração, gerenciamento e migração de autoridades certificadoras em ambientes microsoft podem ser verificados através dos links abaixo:

http://www.scottfeltmann.com/index.php/2010/03/02/move-root-ca-from-w2k3-to-w2k8/

http://technet.microsoft.com/en-us/library/cc772393%28WS.10%29.aspx

http://www.ipsure.com/blog/2010/installation-and-configuration-of-active-directory-certificate-services-on-windows-server-2008-r2-2/

http://technet.microsoft.com/pt-br/library/dd379529%28WS.10%29.aspx

Foi configurada uma CRL (*Certificate Revocation List*), a qual é uma lista de todos os certificados emitidos pela CA que foram revogados. CRLs são o único método de cancelar um certificado antes de sua data de expiração. Para que uma CRL seja eficaz ela deve ser descarregada e instalada em todos os sistemas e aplicações que fazem parte da PKI.

Os certificados serão revogados pela CA quando:

- O utilizador do certificado cessou a sua relação com a organização;
- O utilizador já não quer o certificado;
- A chave confidencial foi perdida ou comprometida;
- A informação no certificado é errada ou inexata;
- O utilizador não está de acordo com as regras da política do CA;
- O sistema para que o certificado foi emitido foi removido.

As partes confiantes (pessoas e organizações usando os certificados para finalidades de autenticação) devem descarregar a CRL pelo menos uma vez por dia e executar as suas limitações ao validar certificados. Esta é a única maneira de se assegurar que um certificado revogado seja recusado realmente em toda parte.

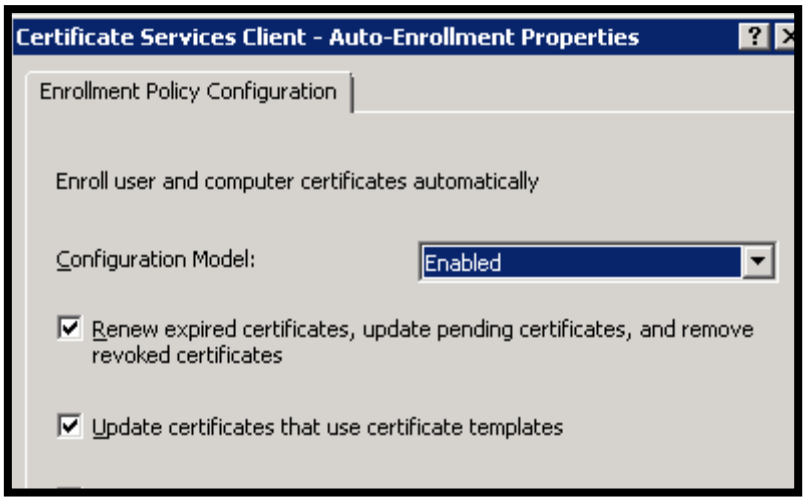
Maiores detalhes sobre CA e revogação de certificados podem ser obtidos nos links abaixo;

http://technet.microsoft.com/en-us/library/cc700843.aspx

Foi ajustado as Group Policys do AD (*Active Directory)* para distribuição de certificados automaticamente, conforme link abaixo:

http://technet.microsoft.com/pt-br/library/dd379529%28WS.10%29.aspx

http://technet.microsoft.com/en-us/library/cc770413%28WS.10%29.aspx



Certificados automaticos por Group Policy

5.2 CONFIGURAÇÃO DE 802.1X NOS CLIENTES

Foi necessário ajustar extensões do *Active Directory* para permitir a configuração automatica do protocolo 802.1X em clientes Windows XP, Windows 7 e Windows 8. Para maiores detalhes acessar o link abaixo:

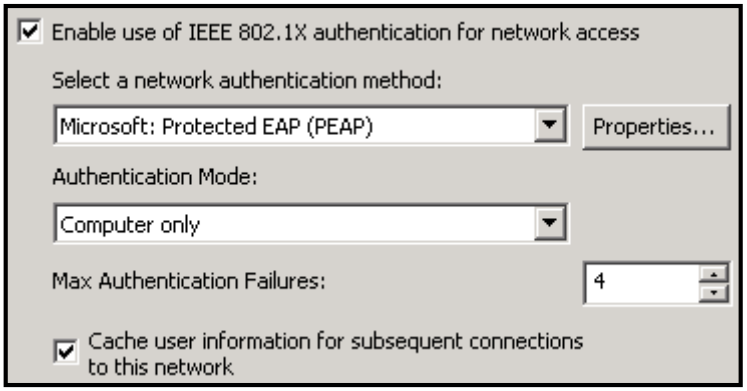
http://technet.microsoft.com/pt-br/library/bb727029.aspx

Após ajuste nova configuração ficou disponível para utilização, foi configurado *Group Policy* para habilitar autenticação 802.1X nos clientes com Windows XP, Windows 7 e Windows 8. Maiores detalhes podem ser obtidos nos links abaixo:

http://technet.microsoft.com/en-us/library/cc730878%28WS.10%29.aspx

http://technet.microsoft.com/en-us/library/dd379919%28WS.10%29.aspx

http://technet.microsoft.com/en-us/library/dd283062%28WS.10%29.aspx



802.1X através de Group Policy em Windows XP, Windows 7 e Windows 8

Após distribuidas as políticas do AD para 802.1X, foi habilitada essa funcionalidade no switch de borda e nas portas do mesmo, conforme procedimentos de configuração a serem vistos posteriormente. Também foi feita a configuração de autenticação por MAC Address para dispositivos como impressora, cameras, condins e entre outros.

Instalado e configurado um servidor NPS com serviço de RADIUS Proxy habilitado, repassando requisições de autenticação para outro servidor IAS com AD. Os dois servidores são baseados no microsoft Windows 2008, estes adicionam uma camada de segurança.

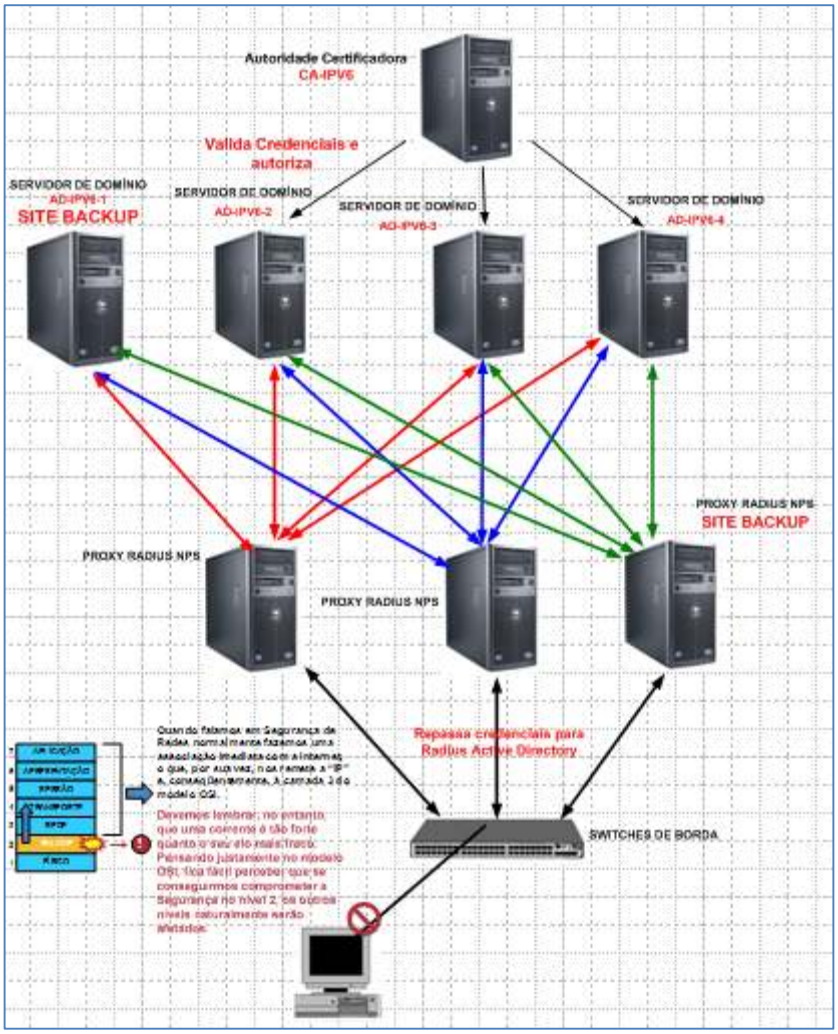
Maiores detalhes podem ser obtidos nos links abaixo:

http://technet.microsoft.com/en-us/network/bb629414.aspx

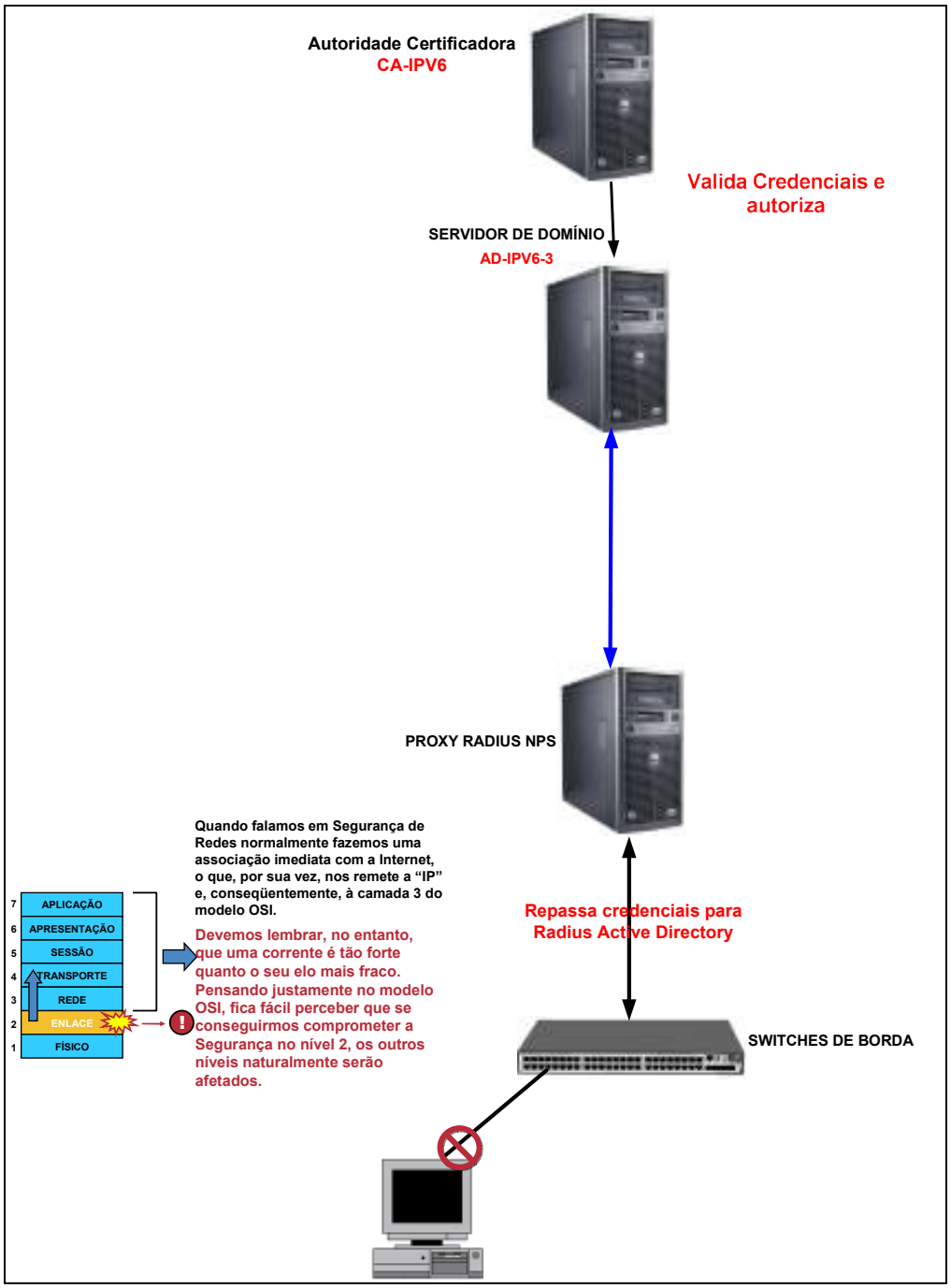
http://www.windowsnetworking.com/articles_tutorials/Understanding-new-Windows-Server-2008-Network-Policy-Server.html

5.3 CONFIGURAÇÃO DE 802.1X NOS CLIENTES

5.4 DESENHO RESUMIDO DA SOLUÇÃO EM UMA ORGANIZAÇÃO



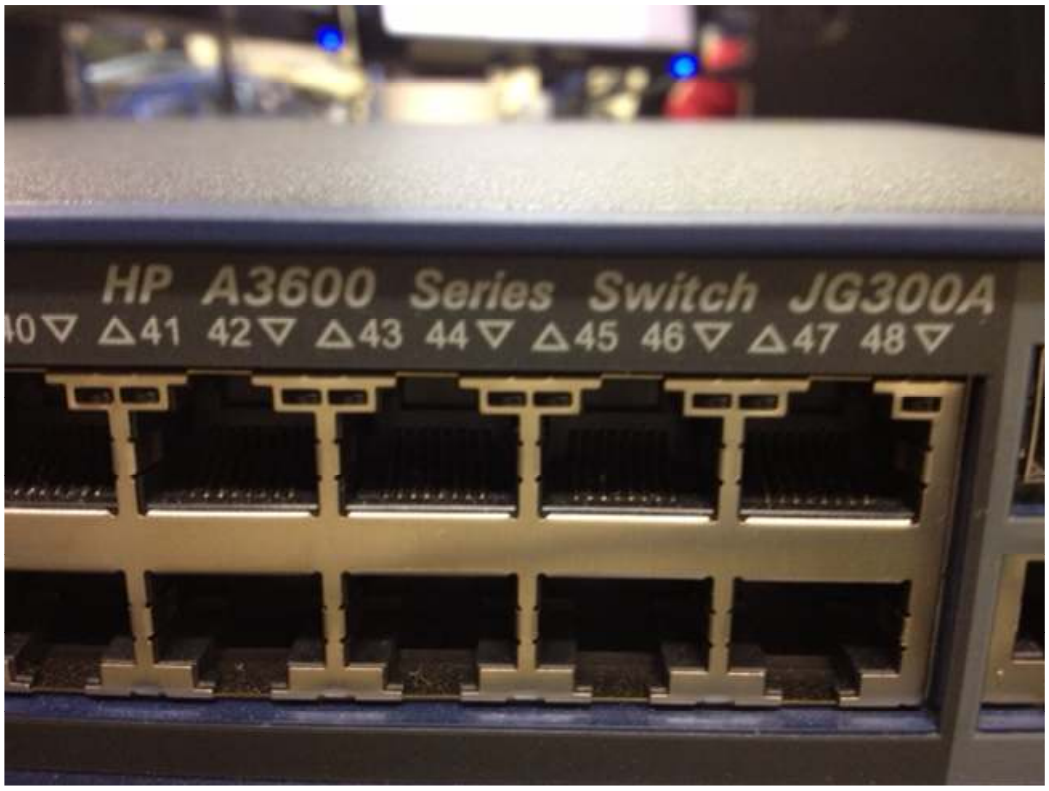
5.5 DESENHO RESUMIDO DA SOLUÇÃO NO LABORATÓRIO



6 PROCEDIMENTOS PARA IMPLEMENTAÇÃO NA REDE DE LABORATÓRIO

6.1 CONFIGURAÇÃO DO SWITCH

Utilizamos no laboratório um switch HP A3600 Series JG300A – 48 portas.



GtkTerm

File Configuration Control signals View

```
 dns domain ipv6.local
#
 ipv6
#
 telnet server enable
#

 port-security trap addresslearned
 port-security trap intrusion
 port-security trap dot1xlogon
 port-security trap dot1xlogoff
 port-security trap dot1xlogfailure
 port-security timer autolearn aging 20
#
 dot1x timer tx-period 10
 dot1x timer supp-timeout 5
 dot1x retry 5
 dot1x timer handshake-period 5
 dot1x timer reauth-period 7200
 dot1x authentication-method eap
#
 mac-authentication user-name-format mac-address with-hyphen
#
 mirroring-group 1 local
#
 password-recovery enable
#
 system-failure maintain
#
acl number 3997
```

/dev/ttyS0 : 9600,8,N,1 DTR RTS

GtkTerm

File Configuration Control signals View Help

```
radius scheme system
 primary authentication 127.0.0.1 1645
 primary accounting 127.0.0.1 1646
 user-name-format without-domain
radius scheme radiusipv6
 server-type extended

 primary authentication ipv6 2001:0db8:cafa:0001::0003
 primary accounting ipv6 2001:0db8:cafa:0001::0003
 key authentication cipher $c$3$X1Lsu5ZEMzzolCX9/rQEbb0B78aEi18=
 key accounting cipher $c$3$xVC0bsbJ/q/6z+vxNanrylpDQPp9/qo=
 timer response-timeout 5
 user-name-format without-domain
 retry 5
#
domain ipv6.local
 authentication default radius-scheme radiusipv6
 authorization default radius-scheme radiusipv6
 accounting default radius-scheme radiusipv6
 access-limit enable 30
 state active

 idle-cut enable 20 10240
 self-service-url disable
  ---- More ----
```

/dev/ttyS0 : 9600,8,N,1 DTR RTS CTS CD DSR RI

GtkTerm

File Configuration Control signals View Help

```
 dot1x guest-vlan 1
 dot1x port-method portbased
#
interface Ethernet2/0/22
 port link-mode bridge
 port access vlan 333

 port multicast-vlan ipv6 333
 stp edged-port enable
 port-security max-mac-count 1
 dot1x re-authenticate
 dot1x max-user 1
 dot1x guest-vlan 1
 dot1x mandatory-domain ipv6.local
 dot1x port-method portbased
 dot1x
#
interface Ethernet2/0/23
 port link-mode bridge
 port multicast-vlan ipv6 333

 dot1x guest-vlan 1
 dot1x port-method portbased

[SWIPV6]
```

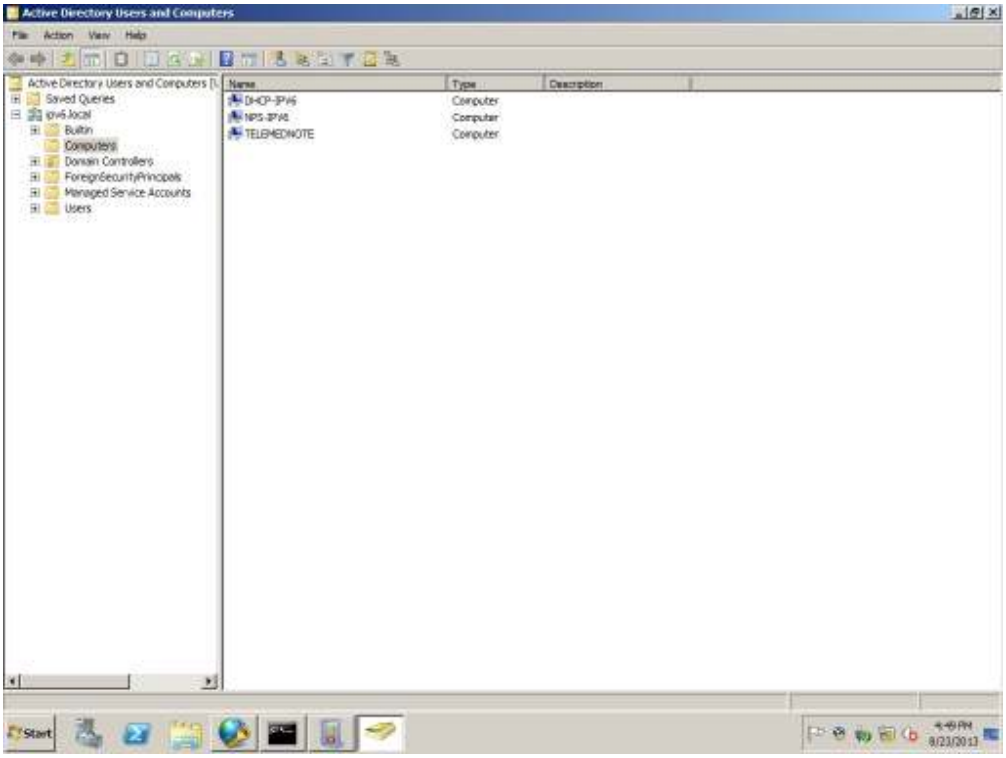
/dev/ttyS0 : 9600,8,N,1 DTR RTS CTS CD DSR RI

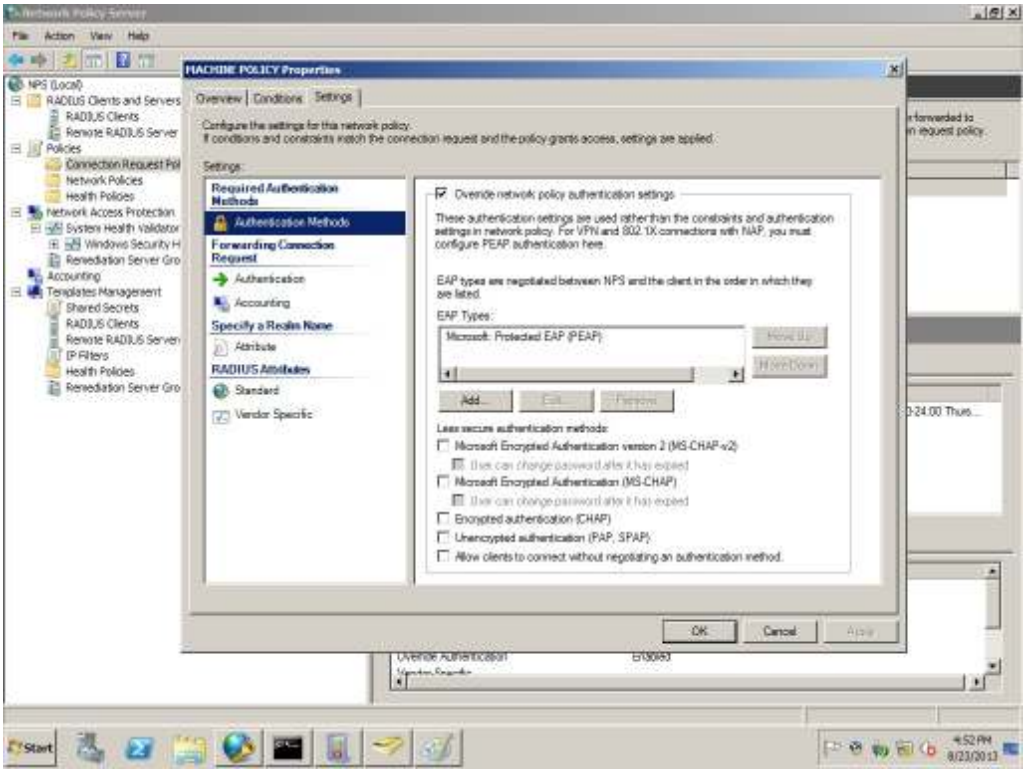
6.2 CONFIGURAÇÃO DA PORTA DO SWITCH:

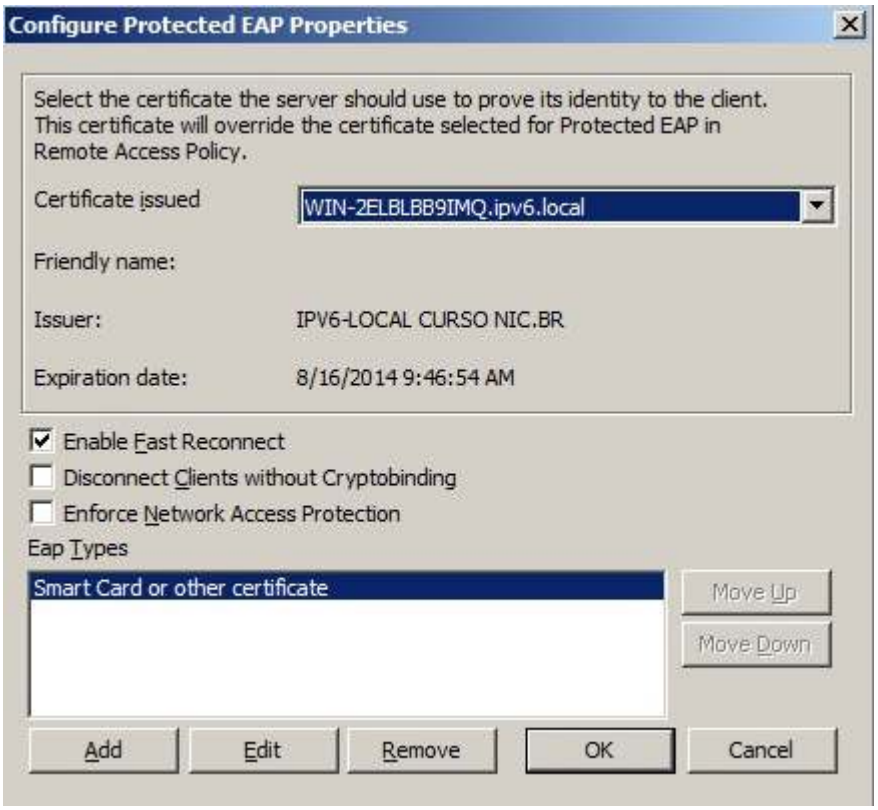
```
[SWIPV6-Ethernet2/0/22]display th
#
interface Ethernet2/0/22
 port link-mode bridge
 port access vlan 333

 port multicast-vlan ipv6 333
 stp edged-port enable
 port-security max-mac-count 1
 dot1x re-authenticate
 dot1x max-user 1
 dot1x mandatory-domain ipv6.local
 dot1x
#
return
[SWIPV6-Ethernet2/0/22]
```

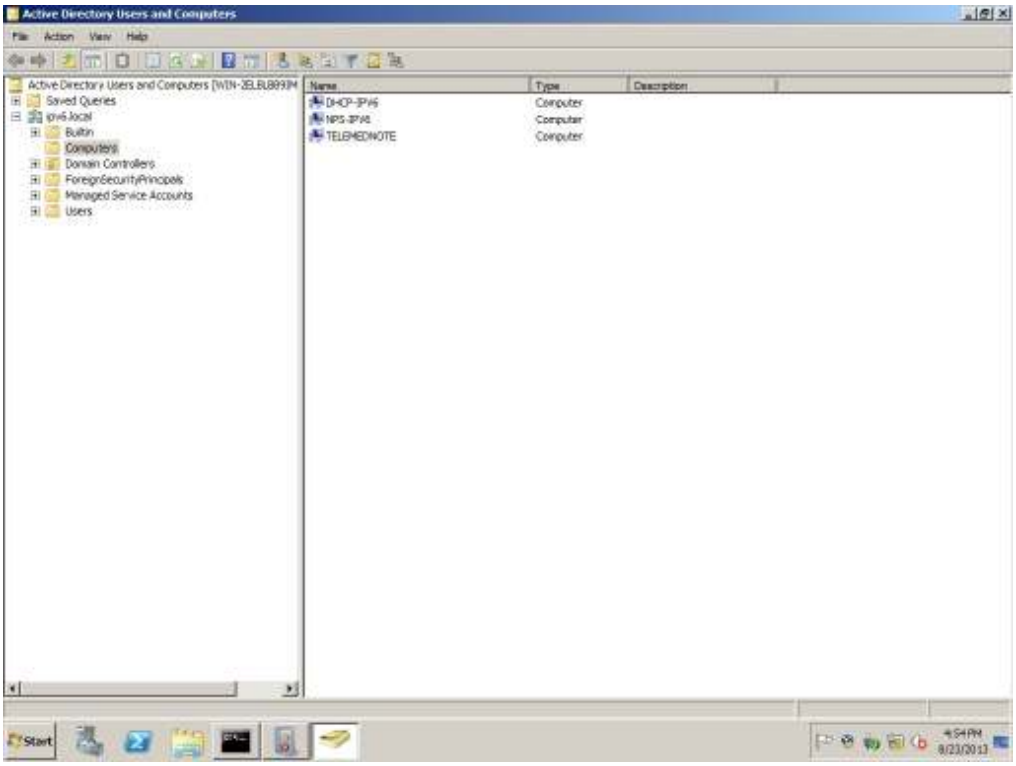
```
802.1x is enabled on port Ethernet2/0/22.
[SWIPV6-Ethernet2/0/22]
```

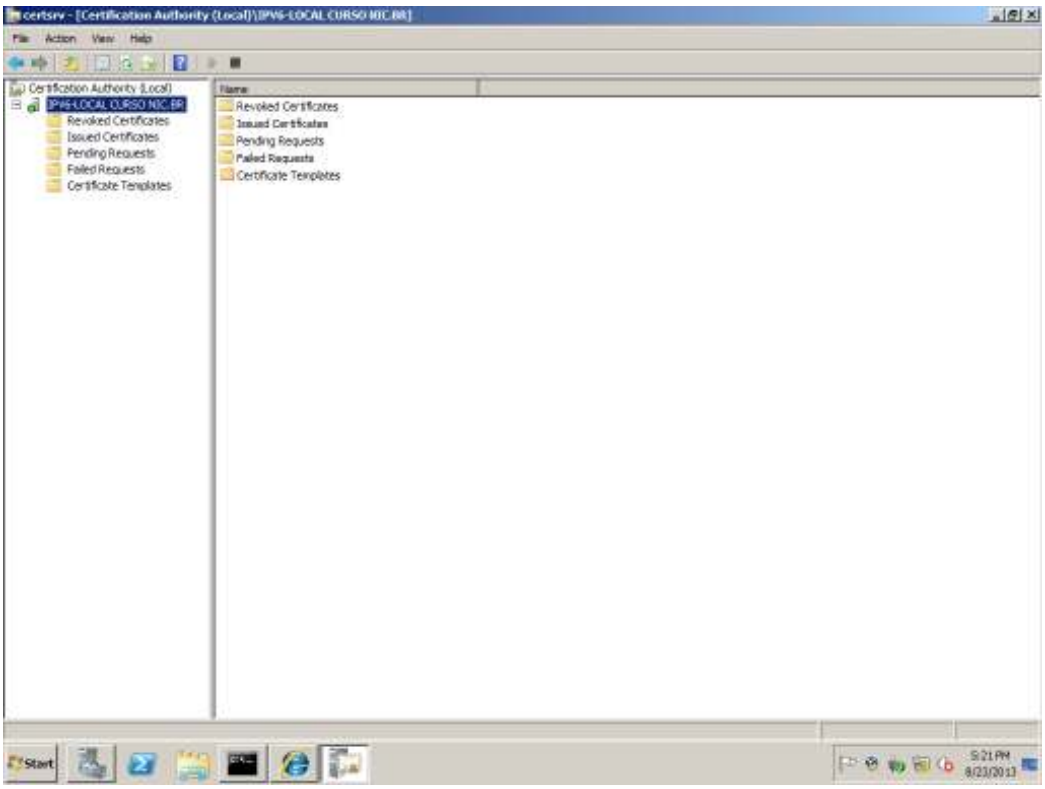
6.3 CONFIGURAÇÃO DOS SERVIDORES

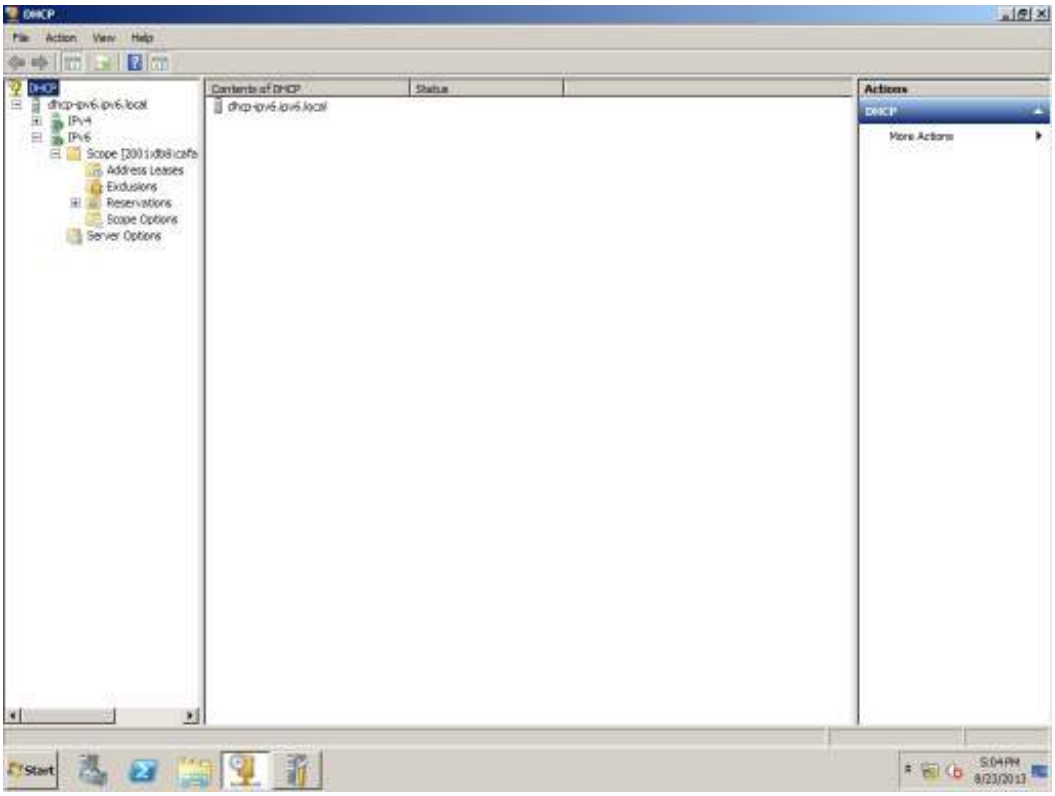


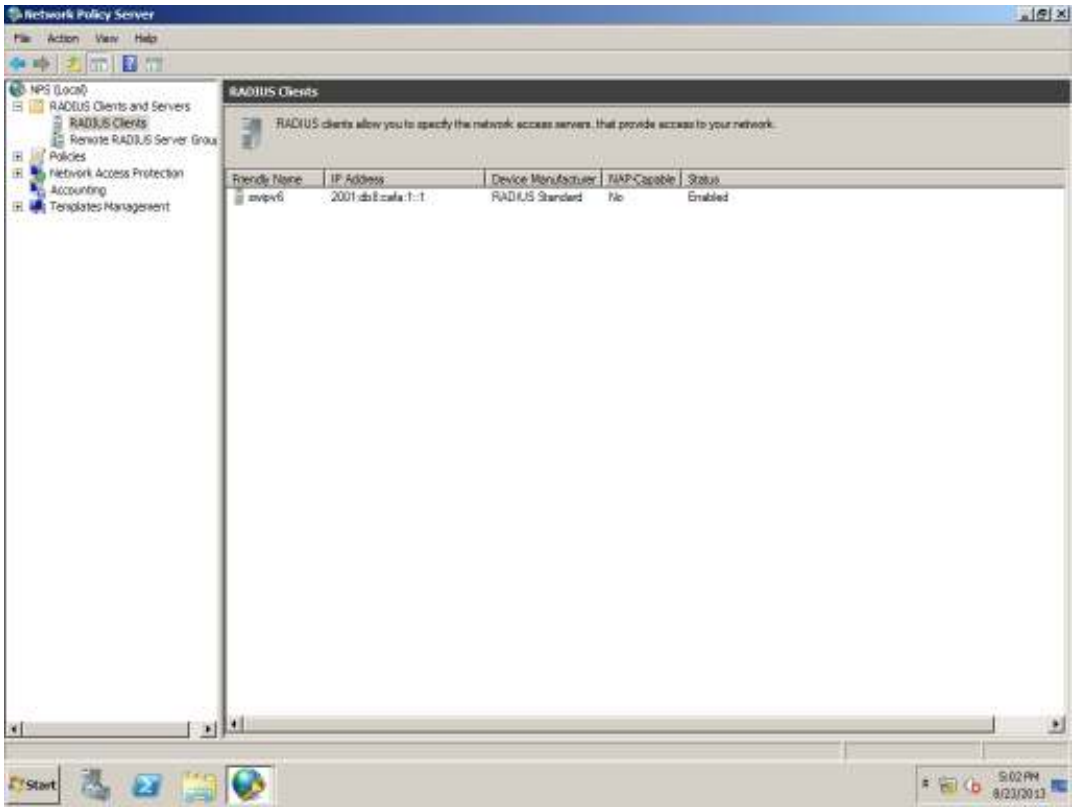

MACHINE POLICY Properties
Overview
Conditions
Settings
Configure the settings for this network policy.
If conditions and constraints match the connection request and the policy grants access, settings are applied.
Settings:
Required Authentication Methods
Authentication Methods
Forwarding Connection Request
Authentication
Accounting
Specify a Realm Name
Attribute
RADIUS Attributes
Standard
Vendor Specific
Override network policy authentication settings
These authentication settings are used rather than the constraints and authentication settings in network policy. For VPN and 802.1X connections with NAP, you must configure PEAP authentication here.
EAP types are negotiated between NPS and the client in the order in which they are listed.
EAP Types:
Microsoft: Protected EAP (PEAP)
Move Up
Move Down
Add...
Edit...
Remove
Less secure authentication methods:
Microsoft Encrypted Authentication version 2 (MS-CHAP-v2)
User can change password after it has expired
Microsoft Encrypted Authentication (MS-CHAP)
User can change password after it has expired
Encrypted authentication (CHAP)
Unencrypted authentication (PAP, SPAP)
Allow clients to connect without negotiating an authentication method.
OK
Cancel
Apply

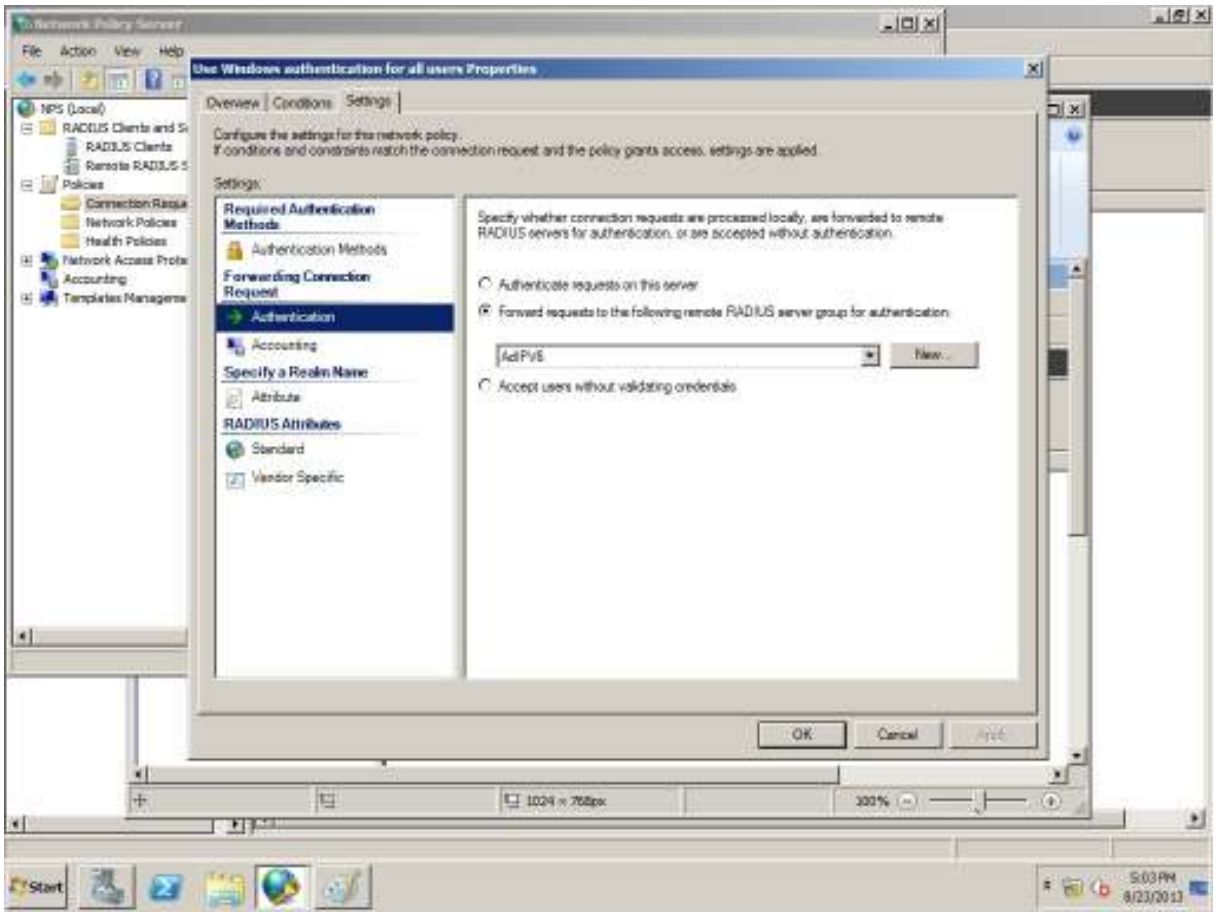

Configure Protected EAP Properties
Select the certificate the server should use to prove its identity to the client. This certificate will override the certificate selected for Protected EAP in Remote Access Policy.
Certificate issued
WIN-2ELBLBB9IMQ.ipv6.local
Friendly name:
Issuer:
IPV6-LOCAL CURSO NIC.BR
Expiration date:
8/16/2014 9:46:54 AM
Enable Fast Reconnect
Disconnect Clients without Cryptobinding
Enforce Network Access Protection
Eap Types
Smart Card or other certificate
Move Up
Move Down
Add
Edit
Remove
OK
Cancel


Active Directory Users and Computers
File Action View Help
Active Directory Users and Computers [WIN-3ELBJ893M
Saved Queries
ipv6.local
Builtin
Computers
Domain Controllers
ForeignSecurityPrincipals
Managed Service Accounts
Users
Name
Type
Description
DHCP-IPV6
Computer
NPS-IPV6
Computer
TELEMEDNOTE
Computer
Start

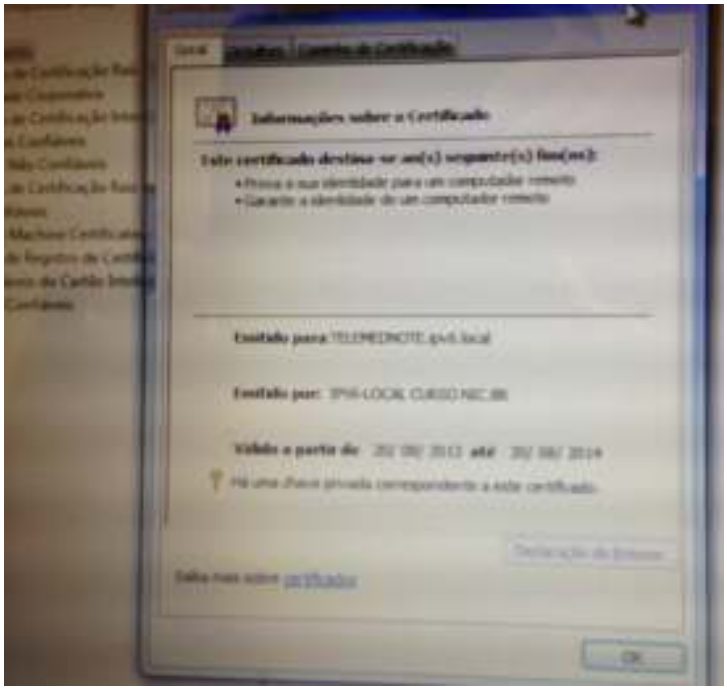

File Action View Help
Certification Authority (Local)
IPV6-LOCAL CURSO NIC.BR
Revoked Certificates
Issued Certificates
Pending Requests
Failed Requests
Certificate Templates
Name
Revoked Certificates
Issued Certificates
Pending Requests
Failed Requests
Certificate Templates
Start

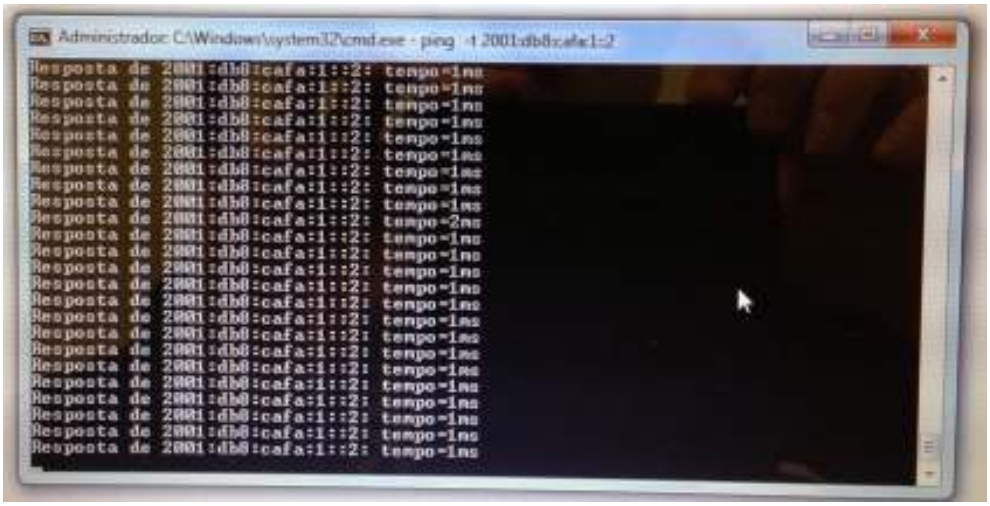

DHCP
File Action View Help
DHCP
dhcp-ipv6.ipv6.local
IPv4
IPv6
Scope [2001:db8:cafe
Address Leases
Exclusions
Reservations
Scope Options
Server Options
Contents of DHCP
Status
dhcp-ipv6.ipv6.local
Actions
DHCP
More Actions
Start
5:04 PM
8/23/2013


Network Policy Server
File Action View Help
NPS (Local)
RADIUS Clients and Servers
RADIUS Clients
Remote RADIUS Server Grou
Policies
Network Access Protection
Accounting
Templates Management
RADIUS Clients
RADIUS clients allow you to specify the network access servers that provide access to your network.
Friendly Name
IP Address
Device Manufacturer
NAP-Capable
Status
ovipv6
2001:db8:cafe:1::1
RADIUS Standard
No
Enabled
Start
5:02 PM
8/23/2013



6.4 CERTIFICADO DO CLIENTE E CONECTIVIDADE




Administrador: C:\Windows\system32\cmd.exe - ping -t 2001:db8:cafa:1::2
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=2ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms
Resposta de 2001:db8:cafa:1::2: tempo=1ms

REFERÊNCIAS

HP A3600 v2 Switch Series. Fundamentals Configuration Guide. Disponível em: <http://h20566.www2.hp.com/portal/site/hpsc/template.BINARYPORTLET/public/kb/docDisplay/resource.process/?spf_p.tpst=kbDocDisplay_ws_BI&spf_p.rid_kbDocDisplay=docDisplayResURL&javax.portlet.begCacheTok=com.vignette.cachetoken&spf_p.rst_kbDocDisplay=wsrp-resourceState%3DdocId%253Demr_na-c03202239-2%257CdocLocale%253Den_GB&javax.portlet.endCacheTok=com.vignette.cachetoken> Acesso em: agosto de 2013.

HP A3600 v2 Switch Series. Security Configuration Guide. Disponível em: <http://h20566.www2.hp.com/portal/site/hpsc/template.BINARYPORTLET/public/kb/docDisplay/resource.process/?spf_p.tpst=kbDocDisplay_ws_BI&spf_p.rid_kbDocDisplay=docDisplayResURL&javax.portlet.begCacheTok=com.vignette.cachetoken&spf_p.rst_kbDocDisplay=wsrp-resourceState%3DdocId%253Demr_na-c03202250-2%257CdocLocale%253Den_GB&javax.portlet.endCacheTok=com.vignette.cachetoken>

IEEE.Org. (IEEE 802.1X, 2001). ANSI/IEEE Standard 802.1X. **Port-Based Network Access Control.** Disponível em: <http://standards.ieee.org/getieee802/download/802.1X-2001.pdf>. Acesso em: agosto de 2013.

RFC 2138 (1997) Request for Comments: Remote Authentication Dial In User Service (RADIUS). Disponível em: <http://www.faqs.org/rfcs/rfc2138.html>. Acesso em: agosto de 2013.

RFC 2284. (1998). Request for Comments: PPP Extensible Authentication Protocol (EAP). Disponível em: <http://www.faqs.org/rfcs/rfc2284.html>. Acesso em: agosto de 2013.

RFC 2460 (1998) Request for Comments: Internet Protocol, Version 6 (IPv6). Disponível em: < http://www.ietf.org/rfc/rfc2460.txt>. Acesso em: agosto de 2013.

RFC 2716 (1999) Request for Comments: PPP EAP TLS Authentication Protocol. Disponível em: <http://www.faqs.org/rfcs/rfc2716.html>. Acesso em: agosto de 2013.