# SOLUÇÃO PROPOSTA DE AUDITORIA DE SEGURANÇA EM FIREWALLS CISCO PARA EMPRESAS COM GRANDE QUANTIDADE DE EQUIPAMENTOS DE SEGURANÇA EM REDE

Proposed solution in safety audit firewalls Cisco for companies with large amount of equipment networking security

**MACIEL FILHO, Marcus Antonius Gadelha**
Faculdade Politécnica de Campinas

**MATTOS, Amarildo Martins de**
Faculdade Politécnica de Campinas

**Resumo***:* Auditoria de Segurança para *firewalls* é algo realmente importante que precisa ser feito em todas as grandes empresas. Este trabalho mostra como fazer um processo automático de Auditoria de Segurança para *firewalls*, os conceitos de segurança, ameaças e vulnerabilidades envolvidas e como é realizado esse processo.

**Palavras-chave:** auditoria, Cisco, firewall, segurança.

**Abstract:** Security Audit for firewalls is a really important activity that need to be performed by all big companies. This work shows how to do an automated Firewall Security Audit and explain the concepts of security, threats and vulnerabilities and how is the process for a Firewall Security Audit.

**Keywords:** audit, Cisco, firewall, security.

## 1. Introdução

Com o passar dos anos, a segurança de redes e sistemas se tornou uma grande necessidade para todas as empresas de grande porte. Com a expansão das atividades ilegais envolvendo acessos não autorizados à sistemas e redes de terceiros, tornou-se indispensável o uso de equipamentos

conhecidos como *firewalls* em quaisquer redes de empresas de grande, médio e pequeno portes.

Como definição um *firewall* é uma parte de uma rede ou sistemas de computadores desenvolvidos para bloquear acessos não autorizados, ao mesmo tempo em que permite o acesso de comunicações autorizadas no meio. O mesmo pode ser descrito como um dispositivo ou conjunto de dispositivos, configurados para permitir, negar, criptografar, decriptografar, ou analisar o tráfego de entrada e saída de dados dos computadores entre diferentes domínios de segurança, sempre baseados em um conjunto de regras específicas (Figura 1).

Seria muito difícil garantir a segurança de qualquer rede sem o uso de um sistema de *firewall*. Uma rede protegida por um *firewall* bem configurado permitirá que apenas os pacotes de dados legítimos (i. é: previamente aprovados pelo mesmo) possam trafegar pela rede. Além da confirmação periódica do fluxo de pacotes, o sistema precisa efetuar o registro das conexões e assim garantir o funcionamento dos diversos fatores relacionados à segurança da rede.

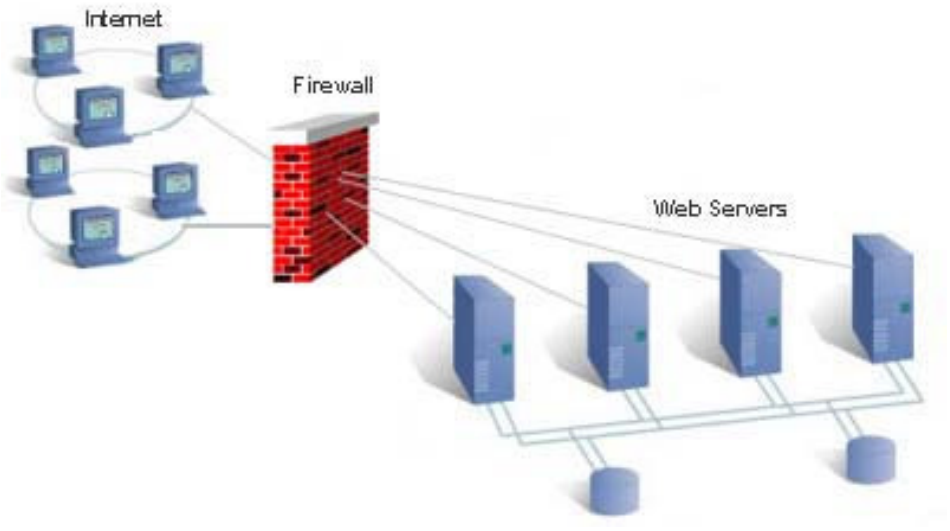



Figura 1 – Representação esquemática de um Firewall em execução em uma rede (ilustração do fastserve.net).

## A. Motivação

Grande parte do trabalho realizado em processos de auditoria de segurança de *firewalls* é feito de forma manual. Este trabalho pode até ser aceitável caso seja feito em uma empresa de pequeno porte que possua poucos equipamentos de rede e pouquíssimos *firewalls* implementados. Mas, para empresas de grande porte, que possuem milhares de equipamentos de rede e *firewalls* ativos (e muitas das vezes em redes continentais), o processo manual de auditoria é completamente inviável, pois exige um grande número de profissionais realizando verificações manuais nos sistemas, a todo o momento.

O uso de profissionais para esse tipo de trabalho contribui consideravelmente para o aumento das taxas de erro nos processos de controle e manutenção de uma rede, além de ser o responsável pela redução da produtividade de um setor ou mesmo da empresa. O desvio de profissionais qualificados para uma única função rotineira acarreta em problemas de gerenciamento, inflacionamento de quadro funcional, e a consequente perda de produtividade de todo um setor estratégico. A automatização dos processos de auditoria de redes de computadores, reduz consideravelmente as taxas de erros humanos, otimiza o uso de profissionais no setor e, garante a rapidez e a precisão com que as verificações são realizadas, e os resultados para análise são obtidos.

## B. Objetivos

O objetivo deste projeto é demonstrar a possibilidade de automação de grande parte dos processos de auditoria de *firewalls* Cisco[7], com o uso da ferramenta apresentada neste documento – o DBPortal. O DBPortal é um sistema de automação voltado para a auditoria de *firewalls* Cisco, que foi inteiramente desenvolvido pelo autor deste trabalho. O uso dessa ferramenta em grandes redes empresariais aumenta a produtividade dos funcionários da

[7] A Cisco, ou *Cisco Systems Inc*. é uma empresa que atua no mercado no desenvolvimento e

área de segurança, permitindo que os mesmos possam executar em paralelo outras atividades de igual importância, nas quais seja necessária uma maior interação humana. Esses profissionais também poderão estar mais envolvidos com a apresentação dos referidos estudos de processos de auditoria, analises de vulnerabilidade e de ameaças, além de um breve estudo no conceito de segurança setorial e global nas empresas em que trabalham.

### C. Contribuições

O DBPortal torna a atividade diária das equipes de segurança um processo mais rápido, preciso e dinâmico, além de garantir uma grande redução nas taxas de erros humanos em atividades de auditoria. Essas ações permitem um aumento na qualidade das informações usadas em auditorias de segurança em *firewalls*, além de permitir que as mesmas possam ser obtidas em um menor intervalo de tempo.

## 2. Segurança da Informação em firewalls

A Segurança da Informação é um tema muito discutido e de extrema importância para a nossa sociedade. No mundo em que vivemos, encontram-se pessoas mal intencionadas (indivíduos, empresas ou mesmo governos), que utilizam de meios e métodos ilegais para adquirir todo tipo de informações. Na maioria das vezes, a intenção dos criminosos é o lucro, através do roubo de informações sigilosas, ou projetos de tecnologia, que possuam grande valor econômico e/ou estratégico, para uma empresa ou mesmo uma nação. Cada empresa precisa estar preparada para poder se defender de forma adequada. Entretanto, devido à capacidade dos *firewalls* – de permitir ou negar qualquer tipo de conexão independente de sua origem – a segurança das informações se tornou um processo um pouco mais fácil de ser administrado.

---

comercialização de equipamentos de rede como roteadores, *switches* e *firewalls*.

Quando se fala em Segurança da Informação a mesma refere-se à tomada de ações necessárias para a garantia da confidencialidade, integridade, disponibilidade, e a todos os demais aspectos de segurança da informação dentro das necessidades do cliente [2]. Devido ao conceito de segurança ser considerado global, o mesmo pode ser adaptado as necessidades de segurança em *firewalls*. Sendo assim, os seguintes itens são definidos:

1) Confidencialidade: refere-se à capacidade que um *firewall* tem em permitir que os usuários acessem determinadas informações, sem que o mesmo conteúdo seja interceptado por outros usuários.
2) Integridade: refere-se à capacidade de um *firewall* em garantir a integridade e veracidade dos dados recebidos.
3) Disponibilidade: refere-se à capacidade dos *firewalls* em relação a disponibilidade de acesso. Acesso esse realizado por todos que precisem do dispositivo para a realização dos objetivos da empresa. Sendo mais específico, a disponibilidade refere-se a capacidade de permitir ou de bloquear um acesso em uma rede.

Alem desses três aspectos principais de segurança em *firewalls*, ainda existem os seguintes pontos [2]:

1) Autenticação: o objetivo da autenticação é garantir que um indivíduo confirme sua identidade como usuário na rede.
2) Não-repudio: ter uma prova de que toda ação executada por qualquer usuário possa ser verificada posteriormente.
3) Legalidade: o *firewall* deve seguir as regras definidas pelo órgão responsável pela segurança na empresa.
4) Privacidade: o *firewall* deve garantir que os tráfegos autorizados possam ser executados de forma anônima, a fim de garantir a privacidade do usuário. Este processo envolve o anonimato do tráfego de informações referentes a avaliações de funcionários, troca de informações entre setores, e os serviços prestados pela empresa.
5) Auditoria: o *firewall* deve ser capaz de auditar todo tipo de ação executada por um usuário em uma rede, detectando tentativas de ataque, ou mesmo fraudes realizadas.

## 3. Fundamentos em auditoria de segurança

As empresas possuem grande necessidade de ter e de administrar o fator segurança em seu ambiente de trabalho, além de serem obrigadas a manter o correto funcionamento de seus *firewalls* dentro das redes de suas próprias instalações. Por isso, é imprescindível a implementação de um processo de verificação e confirmação de toda a estrutura de transmissão de dados nas suas redes corporativas.

A função da auditoria de sistemas é promover a adequação, revisão, avaliação e recomendações para o aprimoramento dos controles internos nos sistemas de informações da empresa, bem como avaliar a utilização dos recursos humanos, materiais e tecnologias envolvidas em seu processamento [1].

### A. Processos em auditoria de firewalls

Em empresas de grande porte, a auditoria de segurança em *firewalls* geralmente é dividida em processos que devem ser constantemente verificados. Abaixo são apresentados os processos que estão relacionados com a auditoria de *firewalls* baseados em [2, 3, 4, 5].

1) Verificação de alterações de senhas de sistema a cada X dias: A alteração de senha de sistema é um processo muito importante em auditoria de *firewalls*. O mesmo garante que os usuários com o péssimo hábito de possuir senhas iguais para todos os sistemas da empresa sejam obrigados a alterar suas senhas em períodos determinados pela própria companhia. Além de forçar a alteração de senhas, as empresas ainda podem utilizar métodos de verificação de caracteres, forçando o usuário do sistema a adotar um conjunto de dígitos variados, como o uso de maiúsculas, minúsculas, caracteres especiais, números, além de uma quantidade mínima de caracteres envolvidos na construção de suas

senhas. O sistema pode, inclusive, verificar se a nova senha é igual às X ultimas senhas usadas neste sistema, evitando que o usuário utilize senhas repetitivas na rede da empresa. Obviamente esta verificação só será possível através do uso de determinados métodos de criptografia para as devidas autenticações.

2) Verificação de existência de *logs*[8] para cada usuário em uma faixa de X dias: O registro de *logs* é a parte mais importante em uma auditoria, pois com eles é possível verificar todas as tentativas de ataque que possam ser realizadas no sistema, obter alertas de mau funcionamento do hardware do *firewall* e, ainda, ajudar na solução de problemas referentes ao tráfego. Afinal, a maior parte do trafego de determinados setores das empresas passam por meio de *firewalls,* tornando possível a visualização dos *logs* dessas conexões, e assim, facilitar a analise dos problemas de rede.
3) Analise da configuração básica do *firewall*: Geralmente, a configuração básica dos *firewalls* possui vários serviços habilitados, sendo que grande parte desses mesmos serviços nunca serão utilizados. Para garantir que apenas os serviços que sejam necessários e seguros estejam habilitados é necessário efetuar uma analise inicial de serviços, registrando assim quais poderão ser desabilitados em um *firewall*. Alguns exemplos de serviços que são normalmente encontrados em uma configuração de *firewall*: Telnet (devido a problemas de segurança é recomendado o uso exclusivo do ssh v2, para qualquer conexão realizada ao firewall), http (que geralmente não é usado) e várias outras configurações que podem, de forma desnecessária, afetar a segurança do *firewall*.

Serviços habilitados no *firewall*, mas que estejam inativos como serviços nas empresas, possuem um potencial de risco e de vulnerabilidade muito grande. Serviços habilitados, mas em desuso, não são monitorados, e muito menos atualizados como pacotes, caso uma brecha de segurança seja descoberta pelos seus desenvolvedores. Ao serem descobertas brechas de segurança nestes serviços, o perigo passa de potencial para real. Através de uma vulnerabilidade exposta e não monitorada, atacantes podem se beneficiar

não somente da vulnerabilidade do serviço, mas também da vulnerabilidade da administração de toda a rede da empresa. Pense em serviços não utilizados em seu *firewall* (a exemplo, VoIP). Uma brecha descoberta para este recurso em uma rede sem monitoramento adequado de serviços inativos, pode se tornar um pesadelo. Isso faz com que o *firewall* esteja vulnerável a um serviço que ele não deveria ter. Esse tipo de situação é considerado uma falha no processo de validação da configuração básica de qualquer *firewall*, e em uma auditoria, isso é um sério problema.

4) Analise de regras de acesso do *firewall*: Quando um *firewall* entra em produção em uma rede, o mesmo deve receber as regras de acesso adequadas para a permissão de determinados tipos de tráfegos. Estes tráfegos geralmente são aprovados pelos responsáveis envolvidos no projeto de implantação deste *firewall*, garantindo que apenas o necessário seja permitido trafegar pela rede. Desta forma, a cada auditoria, torna-se necessário a verificação de todas as regras de acesso do *firewall*, a fim de confirmar que apenas o que foi previamente acordado, é o que esta sendo permitido trafegar na rede.
5) Validação de usuários cadastrados no *firewall*: É normal em qualquer empresa a rotatividade de funcionários, seja na contratação, na demissão, ou na transferência de setor na mesma empresa ou unidade. Desta forma, a validação de usuários é fundamental para garantir que apenas os usuários registrados, e em atividades nos seus devidos setores, tenham acesso coordenado ao sistema e a rede. Esta validação impede o acesso de pessoal não-autorizado aos *firewalls* e, consequentemente, garante que não sejam efetuadas alterações nos mesmos, que possam ocasionar impactos de magnitude incalculável dentro da empresa.
6) Gerar e analisar relatórios a partir de *logs*: É muito improvável que uma analise de *log* seja efetuada sem que esteja acontecendo um problema gerador de impacto dentro da empresa. Sendo assim, a geração automática de relatórios – sempre sumarizando os *logs* de forma a facilitar a identificação e analise de problemas – é algo extremamente necessário em uma auditoria de *firewall*. Sem esses relatórios a análise diária de

[8] *Logs* são registros de eventos relevantes que podem ocorrer no firewall.

milhares (ou até mesmo milhões) de linhas de *log*, tornaria o processo impossível para qualquer ser humano (um indivíduo ou mesmo uma equipe inteira).

7) *Backup* de configurações: Assim como um sistema precisa de um *backup* para garantir que qualquer tipo de alteração possa ter um retorno, o *firewall* também necessita de um *backup* – porém apenas de suas configurações. Através da realização de *backups*, todas as alterações feitas nas configurações podem ser analisadas e, em caso de uma falha, as versões podem ser facilmente comparadas entre si, facilitando assim a identificação do problema.

8) Verificação de versão e modelo do *firewall*: Os fabricantes de dispositivos de segurança disponibilizam com freqüência alertas de segurança informando as possíveis vulnerabilidades que podem permitir ataques do tipo DoS[9] ou DDoS[10]. Inclusive, essas vulnerabilidades podem até mesmo garantir acessos não autorizados ao próprio *firewall*. Normalmente esses alertas vêm sempre acompanhados da solução do problema, podendo ser desde a necessidade da desabilitação de um determinado comando, ou até mesmo uma indicação de urgência para uma atualização no sistema operacional do próprio *firewall*. Sendo assim a verificação de versões destes dispositivos deve ser um procedimento freqüente, para assim garantir a segurança da rede como um todo.

## 4. Vulnerabilidades e ameaças

Um dos motivos pelo qual existem os processos de auditoria em empresas é garantir que os *firewalls* da rede estejam sempre seguros contra qualquer tipo de vulnerabilidade e, assim, possam garantir maior segurança para setores potencialmente vulneráveis da empresa.

Vulnerabilidades são falhas de segurança que podem afetar o sistema operacional do *firewall*. Através dessas falhas é possível que o *firewall* seja

[9] DoS é a sigla para Denial of Service, que é um ataque que visa a negação de serviço de um servidor na rede.
[10] DDoS é a sigla para *Distributed Denial of Service*, que é um ataque que visa a negação de serviço de um servidor

afetado a ponto de gerar um mau funcionamento do sistema. O resultado pode ser um grande risco para toda empresa. Essas vulnerabilidades devem ser analisadas como grandes fatores de risco para a segurança de qualquer companhia.

### A. Ataques de negação de serviço

Os ataques de negação de serviço conhecidos como DoS ou DDoS são ataques que possuem o objetivo de tornar a rede ou serviço indisponível. Normalmente, os ataques do tipo DoS e DDoS são realizados através do envio de uma grande quantidade de pacotes UDP para determinados endereços de Internet, sobrecarregando assim toda a disponibilidade de memória e requisição de resposta para os chamados desses dispositivos. Isso faz com que os equipamentos de redes e serviços fiquem indisponíveis para seus usuários. Os pacotes utilizados para esse tipo de ataque são geralmente via protocolo UDP, por não serem orientados à conexão. Isso facilita o mascaramento do endereço de origem em um ataque.

Muitos se perguntam como é possível ocorrer um ataque DDoS. Muito provavelmente ninguém nunca se perguntou sobre o real motivo da existência dos *malwares*. Boa parte desses *malwares* foi feita com o objetivo de criar *Botnets*. Os Botnets são redes de computadores infectados por *malwares*. Estes computadores passam a possuir comportamento de zumbis, aceitando comandos de seus “mestres” (os criadores dos *malwares*). Normalmente os computadores zumbis se conectam a salas de bate-papo em redes de IRC (*Internet Relay Chat)* controladas por senha, em que o “dono” desta rede de computadores infectados é capaz de executar comandos de ataque.

Esses ataques, nada mais são que o disparo de milhares de pacotes provenientes de todos esses computadores infectados, visando um único alvo (um ou mais servidores ou dispositivos em uma rede), com o intuito de sobrecarregá-lo.

---

na rede, só que de forma distribuida.

Mesmo com todos esses recursos, os ataques nem sempre são efetivos. Os ataques do tipo DoS mais complexos envolvem profundo conhecimento dos equipamentos de rede por parte do atacante, muitas vezes por técnicas de engenharia reversa, permitindo assim a exploração de falhas no processamento de protocolos por parte desses dispositivos. Equipamentos como *firewalls* costumam efetuar a analise de pacotes até a camada de aplicação (camada 7 do modelo OSI), o que tornam esses equipamentos mais suscetíveis aos ataques que se beneficiam da má formatação de pacotes. Tudo isso com o intuito gerar indisponibilidade de serviços em uma rede.

**B. Sniffing**

Boa parte dos protocolos que trafegam pela *Internet* não utilizam criptografia, o que os torna mais suscetíveis a ataques do tipo *Sniffing*. O *Sniffing* é um método para analise de redes que envolve o recebimento de pacotes de todas as origens possíveis, e consequentemente, a formatação dos mesmos em um formato legível para o ser humano (*human-readable*). O *Sniffing* é naturalmente uma técnica utilizada para a análise de funcionamento de redes, ou seja, ela é originalmente uma técnica de uso licito. Dentre os diversos serviços que podem ser “sniffados” destacam-se: telnet, ftp, http, pop, smtp e imap. Porém, nem todos os serviços existentes podem ser “sniffados”, basicamente por trabalharem com conteúdo criptografado. Exemplos são: ssh, sftp, https, pop over ssl, smtp over ssl, imap over e ssl. Explicando de uma forma mais simples, através do uso de técnicas de *sniffing* é possível visualizar dados de usuários em trânsito (a ex.: login e senha) numa rede local, para todos os serviços que utilizam de métodos de autenticação sem criptografia.

**C. Man-in-the-Middle**

Este tipo de ataque permite que o atacante seja capaz de visualizar em tempo real todas as atividades que estão sendo executadas pelo cliente responsável por iniciar uma conexão. Literalmente é poder estar no meio do

caminho em uma rede e saber tudo o que acontece em todo o trânsito de pacotes a sua volta. Um exemplo prático desse tipo de ataque é a conexão via cliente a um site de banco. Enquanto um usuário efetua a conexão com seu banco via *Internet*, um atacante realiza um *dns spoofing*[11] e um *arp spoofing*[12], fazendo com que os pacotes que deveriam ir para o site do banco sejam re-direcionados para a maquina do próprio atacante, que fica entre o cliente e o banco.

Na maioria dos casos, o usuário acaba aceitando um certificado gerado pelo atacante, permitindo ao criminoso estabelecer a conexão com o site do banco, via conta da própria vítima. Obviamente, o certificado gerado pelo atacante será reconhecido como inválido, ou não confiável, pelo navegador de Internet da vítima (Firefox, Opera, IE, etc.). Porém, a maioria esmagadora dos usuários que utilizam recursos bancários via Internet não possuem (ou não se preocupam em ter) o mínimo de conhecimento para reconhecer uma ação criminosa deste nível. Mesmo com o certificado inválido aparecendo em sua tela, poucos serão os usuários que suspeitarão estarem sendo vítimas de criminosos e, consequentemente, darão continuidade ao engodo a que estão se submetendo.

Dentre os protocolos mais utilizados para esse tipo de conexão, o ssh versão 1 é considerado bastante vulnerável a este tipo de ataque. Por este motivo, é recomendado apenas a instalação e o uso do protocolo ssh versão 2 em todos os servidores de rede, assim como nos próprios *firewalls* das empresas. O ataque do tipo *man-in-the-middle* pode ser detectado facilmente por sistemas de IPS[13], sendo os mesmos capazes de efetuar o bloqueio da porta utilizada para este ataque. Porém, esse ponto não será abordado neste trabalho.

## D. Port Scan

[11] Ataque usado para direcionar o dns de um host para um destino falso.
[12] Ataque usado para enviar um endereço de hardware falso ao requisitante.
[13] IPS é a sigla em inglês para Intrusion Prevention System.

Este ataque tem como objetivo procurar portas que possam estar abertas em determinados servidores, a fim de gerar algum tipo de ataque em alguns dos serviços encontrados. Em *firewalls* bem configurados, esses ataques podem ser reconhecidos através da análise dos *logs*, pela grande quantidade de acesso negado a determinadas portas para um mesmo endereço IP de origem.

## 5. O Projeto

Devido à ausência de softwares que atendam este ramo de automação, o projeto em discussão, denominado DBPortal, foi desenvolvido com o intuito de automatizar a maior parte dos processos de auditoria de *firewalls*. O DBPortal é um sistema *Web* capaz de realizar, de forma automatizada, grande parte dos processos de auditoria de *firewalls* existentes nas grandes empresas, que normalmente são realizados de forma manual pelo corpo funcional destas mesmas companhias.

O sistema foi desenvolvido e implementado em um servidor DUAL Intel® Xeon™, com CPUs de 3.20GHz, possuindo discos SCSI de alta performance. O sistema operacional instalado é o Red Hat Enterprise Linux (RHEL) AS release 3 (Taroon Update 9). Como ferramentas de desenvolvimento foi utilizado o PHP versão 4.3.2 como linguagem de script para *Web*, e o banco de dados relacional de código aberto MySQL versão 4.1.22. Também foram utilizadas como ferramentas de integração, *scripts* em *Shell* usando Expect, Bash, e AWK, além de diversas outras ferramentas de código aberto disponíveis para o projeto.

O DBPortal conta com uma interface *Web* desenvolvida através de técnicas de IHC[14], facilitando seu aprendizado e uso por parte dos usuários. Toda a tela apresentada pelo sistema é resultado de 2 anos de pesquisa com a colaboração de diversos usuários-testes do sistema, incluindo testes de

[14] IHC é a sigla para Interação Humano Computador.

posicionamento de menus e formulários, além da escolha das cores utilizadas, representado o resultado de funções do sistema.

Apesar do DBPortal já estar em uso, seu desenvolvimento continua com a criação de novos recursos de automatização de processos de segurança, os quais ainda estão sendo efetuados de forma manual nas empresas e, com isso, consumindo uma quantidade desnecessária de tempo e recursos em seus processos. Para que o sistema pudesse efetuar todas suas atividades atualmente implementadas foram criados *scripts* especializados com capacidade de interação com os *firewalls* na rede. Esses *scripts* especializados possuem a função de processamento do retorno de informação gerada pelos *firewalls*, em complemento a já mencionada interface *Web* do sistema.

O DBPortal possui diversas funcionalidades que foram implementadas de forma modular, visando sempre a automatização de um processo de auditoria de *firewalls* dento de uma empresa de grande porte. Para facilitar o completo entendimento de sua estrutura e funcionalidades, primeiro serão apresentados todos os processos independentes de interação com cada dispositivo registrado no sistema. Posteriormente serão abordados os módulos funcionais e o processo de *backup* dos arquivos de configuração dos dispositivos:

**A. Script de captura de configuração**

Este *script* é responsável pela captura da configuração dos *firewalls* que estejam em atividade na rede. Essa captura envolve dados como numero serial, versão de software e modelo do *firewall*. Este *script* foi totalmente escrito em uma linguagem de *script* não muito conhecida – o Expect[15].

A linguagem de *script* Expect é uma ferramenta de teste e automação do Unix, escrita por Don Libes[16], para ser uma extensão da linguagem de *script*

[15] Saiba mais sobre a linguagem de script Expect em `http://expect.nist.gov/`
[16] `http://en.wikipedia.org/wiki/Don_Libes`

Tcl[17]. O expect é destinado para a realização de interação com aplicações como telnet, ftp, passwd, rlogin, tip, ssh, e muitas outras. O *script* criado na linguagem Expect, possui a capacidade de interpretar os resultados recebidos dos outros *scripts* de interação com os *firewalls*. A partir desses dados, o Expect é capaz de descobrir qual comando utilizar no sistema. Esse processo garante a interação do DBPortal com *firewalls* de diferentes modelos e fabricantes, que estejam em funcionamento em uma mesma rede. Esses *scripts* são capazes de acessar o *firewall* independente do tipo de protocolo utilizado (a ex.: ssh ou telnet). Os mesmos possuem um recurso de *time-out* junto ao sistema de investigação, o que lhes permite tentar outros protocolos caso o primeiro não esteja disponível. Os *scripts*, por padrão, sempre procurarão primeiro pelo protocolo ssh em uma rede, sendo capazes de acessar *firewalls* que utilizem os protocolos jumpbox [18] ou mesmo socks[19] para suas conexões. No entanto, para que possam funcionar corretamente, estes métodos de acesso dependem da sua configuração na interface *Web* do DBPortal.

## B. Script de alteração de senhas de console

Independente de seus modelos ou fabricantes, os *firewalls* possuem senhas de emergência que geralmente são utilizadas para os processos de reparação via console, ou mesmo no caso do servidor de autenticação não estar funcionando. O processo de troca de senhas do DBPortal também inclui as senhas de emergência, o que tornou necessário a criação deste *script* para automatizar esta atividade. Tente imaginar um operador (ou mais de um) precisando trocar as senhas de forma manual para todos os *firewalls* das redes que administra, quando esse número ultrapassa a casa do milhar. Esta tarefa hercúlea seria uma atividade que tomaria tempo em demasia, deslocaria muitos funcionários para essa atividade, além de ocasionar a perda de produtividade do setor e, consequentemente, da empresa como um todo. Para

[17] Tcl é uma linguagem de programação para scripts maiores informações em http://pt.wikipedia.org/wiki/Tcl
[18] *Jumpbox* é uma palavra em inglês usada para representar uma máquina que é utilizada para acessar outra máquina.
[19] Socks é um protocolo que facilita o roteamento de pacotes entre cliente e servidor.

tentar otimizar ao máximo essa atividade, foi criado este *script*, que possui as mesmas capacidades do *script* de captura de configuração, porém com o objetivo de trocar as senhas de emergência de todos os dispositivos em análise nas redes.

### C. Scripts de interpretação de retorno dos scripts de coleta de dados

Para que seja possível a interpretação dos resultados recebidos pelos *scripts*, e que estes mesmos resultados possam ser administrados via interface *Web* do DBPortal, foram criados para o sistema diversos outros *scripts* em Bash e AWK. Estes *scripts* extras são capazes de analisar o retorno recebido pelos *scripts* de coleta de dados, além de serem capazes de classificar o processo de captura de configuração, ou alteração de senhas de emergência. Ao avaliar o processo (se foi bem sucedido ou não), a informação é posteriormente armazenada em banco de dados.

### D. Script de controle de processos

Para que seja possível a execução simultânea do sistema para as capturas de configuração, e para a viabilidade das diversas trocas de senha dos dispositivos, foi criado um *script* que controla a inicialização de vários processos simultâneos, sempre limitados por um numero máximo definido em configuração. Desta forma, executando 20 instâncias simultâneas via DBPortal, foi possível efetuar a captura de configuração de 150 *firewalls* por minuto, o que é uma quantidade muito expressiva.

### E. Script de controle de alteração de configuração

No DBPortal cada captura de configuração deve ser classificada quanto a possíveis alterações efetuadas na configuração. É importante para o sistema gerenciar esse tipo de informação, além de ser apto a informar ao operador se

a configuração foi alterada ou não. Para que isso seja possível, o *script* de controle de alteração de configuração recebe o retorno do comando md5sum[20] que efetua a comparação entre a configuração atual e a configuração anterior. Caso ambas sejam idênticas, a nova configuração é descartada. Caso sejam diferentes, o sistema irá registrar a nova configuração, e irá gerar um arquivo com as diferenças entre as duas versões mostrando em vermelho o que foi removido e em azul o que foi adicionado.

Caso a nova configuração recebida seja diferente da configuração anterior, e na configuração da interface *Web* do DBPortal seja informado um email de notificação, será enviado para esse destinatário a nova configuração do *firewall*.

## F. Interface Web

A interface *Web* foi criada utilizando basicamente a linguagem de script PHP e o banco de dados relacional de código aberto MySQL. Para otimizar o sistema com a recarga localizada de novas informações em tela foram implementados alguns recursos em AJAX (*Assincronous JavaScript and XML*). Essa implementação também contribuiu para a melhoria da interface do usuário, tornando-a mais amigável e intuitiva. Através da interface *Web*, o usuário (também chamado de operador) poderá ter o controle total das informações sobre os dispositivos cadastrados. Também poderá efetuar modificações globais ou isoladas nos dispositivos, assim como manter o monitoramento em uma rede global de computadores, tudo isso visando o processo de auditoria de *firewalls*.

Atualmente o DBPortal apresenta os seguintes recursos implementados e completamente funcionais: inventário de equipamentos, *backup* de configurações, alteração de senhas de sistema, controle de quantidade de *logs*, geração de arquivo com diferença entre as configurações, geração de relatório

[20] `md5sum` é um programa capaz de calcular os *hashes* de 128-bits em MD5, como descrito da RFC 1321.O hash MD5 (ou *checksum*) funciona como uma assinatura digital compacta de um arquivo. Por ser uma assinatura de 128-bits será muito difícil encontrar dois md5sum iguais para arquivos diferentes.

de analise de configuração básica, controle de troca de senhas, controle de *healthchecking,* e pesquisa por comandos em configurações:

1) Inventario de equipamentos: Através do inventário de equipamentos, o operador pode obter de forma imediata todas as informações referentes aos dispositivos cadastrados no sistema. Nele é possível encontrar informações como endereço de IP, *hostname*, saber qual cliente que utiliza o equipamento, além de diversas outras informações úteis. As informações contidas no inventário de equipamentos podem ser encontradas pelo sistema de busca criado para o DBPortal. Por exemplo, uma busca por 192.168 no campo de IP, irá retornar todos os endereços de IP que começam com 192.168 (ex.: 192.168.0.1, 192.168.0.2). O sistema de busca ignora capitulação (não reconhece a diferença entre maiúsculas e minúsculas entre os caracteres digitados). Essa implementação favorece encontrar o mesmo alvo em diferentes formatos (a ex.: CISCO, Cisco ou cisco).
2) *Backup* de configurações: O *backup* de configurações é realizado para todos os equipamentos que tenham configurações específicas para a realização de *backup*, como pode ser visto na Figura 2. Este procedimento está implementado no sistema via cron. O mesmo realiza *backups* simultâneos a cada minuto, iniciando diariamente à zero hora (0:00). Todos os equipamentos com a configuração de *backup* habilitada são imediatamente adicionados à fila de execução de *backup*. Além do *backup* diário de todos os *firewalls*, também é possível realizar um *backup* a qualquer momento por intermédio do operador (vide Fig. 3. e Fig. 4).

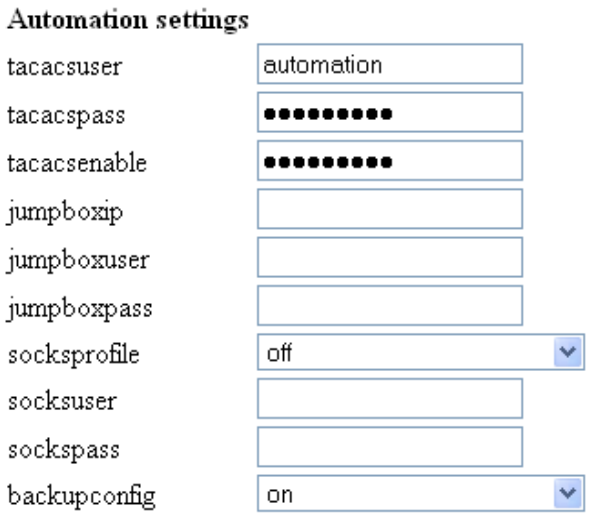



Figura 2 - Demonstra como são os campos de automação no DBPortal.

3) Alteração de senhas de sistema: Esta opção permite que todos os equipamentos com a opção de *backup* de configuração ativa, possam ter suas senhas alteradas pelo DBPortal. Esta opção é muito útil quando se faz necessário efetuar a troca de todas as senhas de sistema de vários equipamentos ao mesmo tempo. Essa automatização contribui para reduzir a zero o problema de erro humano, muito freqüente em tarefas repetitivas. Por ser automatizada, esta tarefa é incomensuravelmente mais rápida que um ser humano. Para se ter uma ideia da performance do sistema de troca de senha, é possível efetuar a troca das senhas de aproximadamente 150 equipamentos por minuto, executando 20 processos simultâneos. O numero de processos deve ser registrado na configuração do *script* de alteração de senhas.

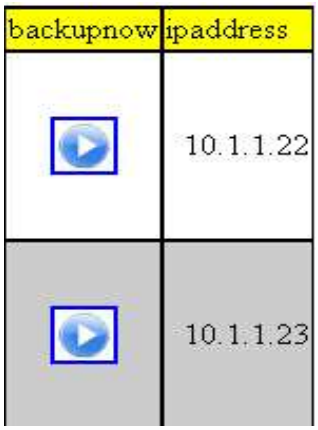



Figura 3 - Demonstra a função de execução de *backup*, sem necessariamente estar na fila de execução.

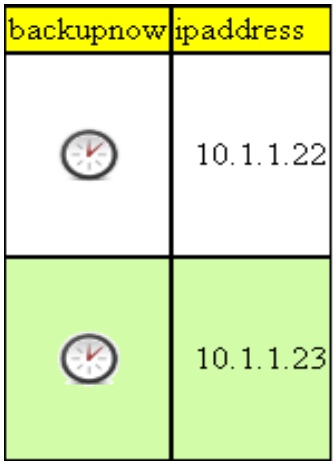



Figura 4 - Demonstra a função de execução de *backup* já inscrita na fila de execução.

4) Controle de quantidade de *logs*: O controle de quantidade de *logs* é um dos processos automatizados que mais beneficiam a infraestrutura local de administração de equipamentos para auditoria. Além de ser automatizada, esta ação exime o operador de possuir conhecimentos avançados em administração de sistemas UNIX via linha de comando, além de tornar desnecessário que o operador efetue o acesso ao servidor de *syslog* do *firewall* de forma manual. Sendo assim, o controle de quantidade de *logs* é executado pelos *scripts* de sistema do próprio DBPortal, que possui chaves RSA com todos os servidores de *syslog* ativos, permitindo assim que o sistema acesse outros servidores no intuito de obter informações específicas como, quantidade de *logs*, tamanho dos *logs*, cabeçalho do primeiro *log* e rodapé do ultimo *log*.

5) Geração de arquivo com diferença entre configurações: Esta função é gerada automaticamente pelo processo de *backup* de configuração, ajudando o operador do sistema saber qual foi a última alteração na configuração de determinado equipamento. Um exemplo pode ser visto na Figura 5.

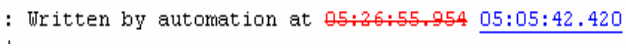
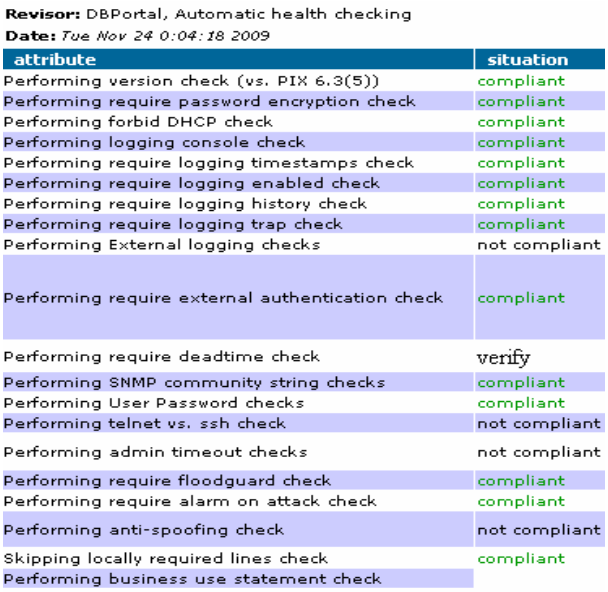



Figura 5 - Demonstra o funcionamento do arquivo de diferença de configurações, gerado pelo processo de *backup* de configurações, sendo vermelho o removido e azul o adicionado na configuração.

6) Geração de relatório de analise de configuração básica: O relatório de configuração básica é executado no momento em que o *backup* de configurações é completado. Este processo é responsável por analisar a configuração básica do equipamento, tornando possível saber se o mesmo está configurado e funcionando de acordo com o que foi previamente estabelecido. Um exemplo deste relatório pode ser visto na Figura 6.



**Revisor:** DBPortal, Automatic health checking
**Date:** *Tue Nov 24 0:04:18 2009*

| attribute | situation |
|---|---|
| Performing version check (vs. PIX 6.3(5)) | compliant |
| Performing require password encryption check | compliant |
| Performing forbid DHCP check | compliant |
| Performing logging console check | compliant |
| Performing require logging timestamps check | compliant |
| Performing require logging enabled check | compliant |
| Performing require logging history check | compliant |
| Performing require logging trap check | compliant |
| Performing External logging checks | not compliant |
| Performing require external authentication check | compliant |
| Performing require deadtime check | verify |
| Performing SNMP community string checks | compliant |
| Performing User Password checks | compliant |
| Performing telnet vs. ssh check | not compliant |
| Performing admin timeout checks | not compliant |
| Performing require floodguard check | compliant |
| Performing require alarm on attack check | compliant |
| Performing anti-spoofing check | not compliant |
| Skipping locally required lines check | compliant |
| Performing business use statement check | |

Figura 6 - Demonstra o relatório de verificação de configuração básica.

7) Controle de troca de senhas: A troca de senhas de sistema é uma atividade que faz parte de qualquer auditoria. Porém a periodicidade da troca de senhas pode variar de acordo com a definição da política de segurança de cada empresa. Sendo assim, o DBPortal possui um campo específico para o cadastro da data de alteração de senha para cada dispositivo, ou conjunto de dispositivos. As senhas podem ser alteradas manualmente, ou de forma automática, através do *script* de alteração de senhas. Este campo possui 3 (três) opções de filtro: vermelho, amarelo e verde. Dependendo do tempo em que foi feita a ultima alteração de senha, o campo recebe uma cor diferente. Os prazos que influenciam a marcação por cores, variam de empresa para empresa e das suas respectivas políticas de monitoramento e auditorias.
8) Controle de *healthchecking*: O *healthchecking* é um processo de confirmação, voltado para a verificação da correta funcionalidade do equipamento em monitoramento. Seu funcionamento é planejado conforme as políticas de segurança de cada empresa. A verificação da configuração básica dos equipamentos faz parte desta atividade, assim como o processo de verificação de quantidade de *logs*. Porém, alguns processos ainda precisam ser executados de forma manual, como a verificação das regras permitidas e negadas pelo equipamento. O DBPortal possui um campo para o cadastro da data em que foi realizado este processo, e também permite que o mesmo seja filtrado em vermelho , amarelo ou verde, facilitando assim saber quais equipamentos tiveram seus *healthchecks* efetuados no período acordado.
9) Pesquisa por comandos em configurações: A função de pesquisa por comandos de configurações é uma das funções que mais facilita a vida das pessoas responsáveis pelas auditorias. Esta função permite que os alertas de segurança dos fabricantes de equipamentos possam ser verificados em todos os dispositivos que possuem o *backup* de configuração ativo. Tudo isso em poucos segundos. Caso este processo fosse efetuado de forma manual, o tempo para sua realização poderia ser contabilizado em dias, meses, ou até anos – tudo dependendo da

quantidade de equipamentos a serem verificados. O funcionamento desta característica é bem simples. Devemos apenas informar o comando a ser procurado, e o retorno será de todos os equipamentos que possuem este comando ativo, além do seu modelo e sua versão de software em uso. A obtenção das informações sobre modelo e versão de software é de extrema importância, visto que nem todos os problemas de segurança registrados atingem (ou podem atingir) todos os modelos e/ou versões de software destes equipamentos.

Agregando todas essas nove funcionalidades, podemos expor a estrutura de operação do DBPortal em três módulos funcionais distintos: inventário, controle e backup (Figuras 7, 8, 9 e 10):

1) Inventário: inventário de equipamentos, alteração de senhas de sistema e, pesquisa por comandos em configurações.
2) Backup: backup de configurações, geração de relatório de analise de configuração básica e, geração de arquivo com diferença entre as configurações.
*3)* Controle: controle de quantidade de logs, controle de troca de senhas e, controle de *healthchecking.*

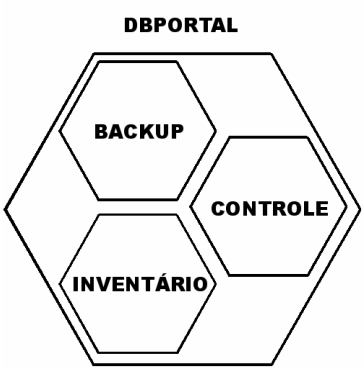



Figura 7 - Apresentação dos três módulos principais do DBPortal.

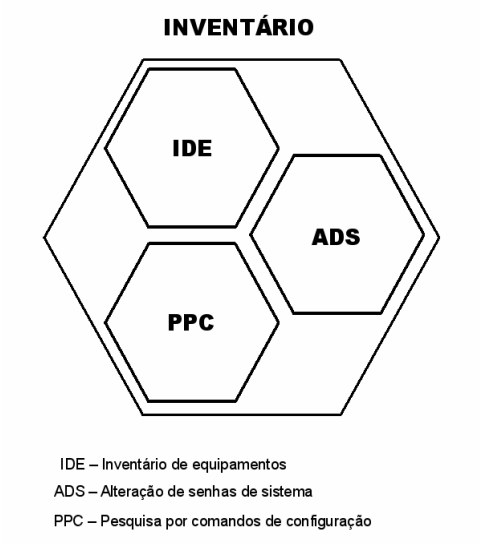



Figura 8 - Módulo de inventário expandido, apresentando seus sub-módulos IDE, ADS e PPC.

## G. Entendendo o processo de Backup

Um dos processos mais interessantes que o DBPortal possui é o *backup* de arquivos de configuração de *firewalls*. Cada dispositivo cadastrado no sistema precisa que um conjunto de duas variáveis estejam ativas para assim permitir que o processo de gravação dos arquivos ocorra. A primeira é a variável backupconfig e a segunda é a variável runnow. Por padrão, ambas as variáveis sempre estarão inativas para o dispositivo cadastrado.

O primeiro passo é ativar a variável backupconfig de cada dispositivo. Por padrão, ao cadastrar qualquer dispositivo no DBPortal, essa variável é marcada como OFF. A marcação desta variável para ON deve ser feita sempre de forma manual por um operador no sistema. O segundo passo é ativar a variável runnow, de 0 (zero) para 1 (um). Esse procedimento pode ser feito tanto de forma manual, quanto de forma automática.

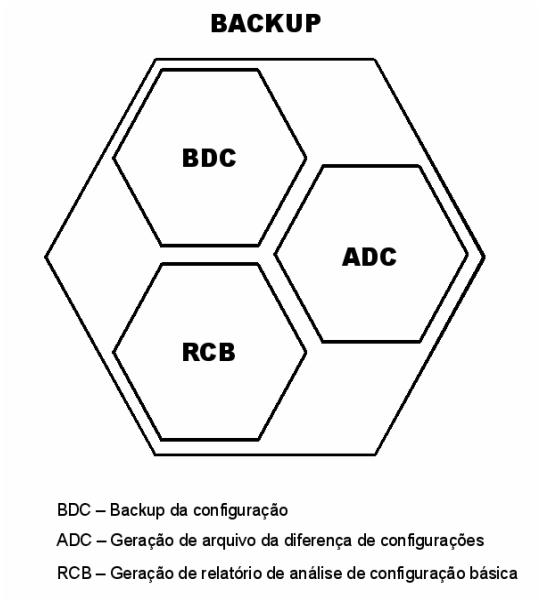



Figura 9 - Módulo de *backup* expandido, apresentando seus sub-módulos BDC, ADC e RCB.

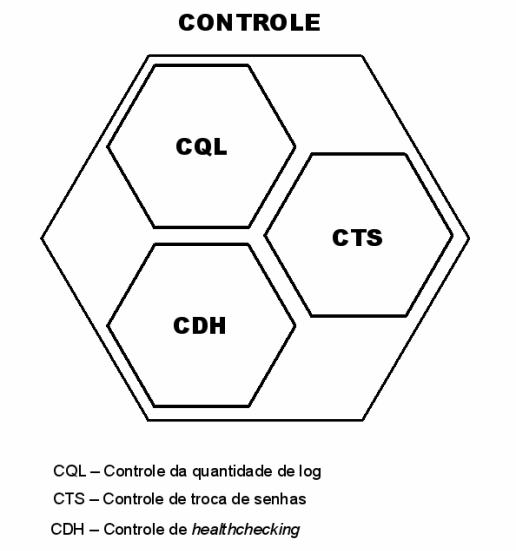



Figura 10 - Módulo de controle expandido, apresentando seus sub-módulos CQL, CTS e CDH.

O DBPortal possui um sistema ativo de *backup* de arquivo de configuração de dispositivos, que é executado de 01 em 01 minuto, de forma contínua e ininterrupta. Mas para que o dispositivo possa ser acessado pelo sistema de *backup*, além da variável backupconfig, a variável runnow também precisa estar ativa.

O processo automático se inicia toda zero hora (0:00) de cada dia, quando o sistema força todos os dispositivos cadastrados que possuam a variável backupconfig em ON, a ter sua variável runnow em 1 (veja Fig. 11)

O processo manual pode ser feito por intermédio de um operador. Um usuário com permissão de alteração das configurações do dispositivo no sistema, pode alterar o valor da variável runnow para 1, de forma manual, efetuando sua ativação. Após essa etapa, o sistema de verificação de dispositivos prontos para *backup* (que está sempre pronto para efetuar *backups* de dispositivos de 01 em 01 minuto) irá identificar o mesmo como READY, iniciando imediatamente o processo de gravação dos seus arquivos de configuração.

A cada minuto o sistema consegue efetuar o *backup* de até 150 dispositivos, de forma simultânea. A quantidade de processos simultâneos através do DBPortal depende exclusivamente da capacidade do servidor onde o sistema está instalado. Basicamente, a capacidade do hardware e a largura de banda da rede.

Após efetuar o *backup*, a variável runnow para aquela máquina voltará a ser marcada como 0. Esse mesmo dispositivo, enquanto estiver com sua variável backupconfig ativa (ON), só poderá passar por um novo processo de *backup*, caso o ciclo de zero hora (0:00) se repita, ou caso neste intervalo de tempo, um operador resolva ativar novamente a variável runwon para esse dispositivo. E durante um dia, de forma manual, uma mesma máquina pode sofrer vários *backups*, se os operadores do sistema assim o desejarem. Tudo depende de seus operadores e da necessidade da empresa com relação aos dados de seus dispositivos em um processo de auditoria padrão.

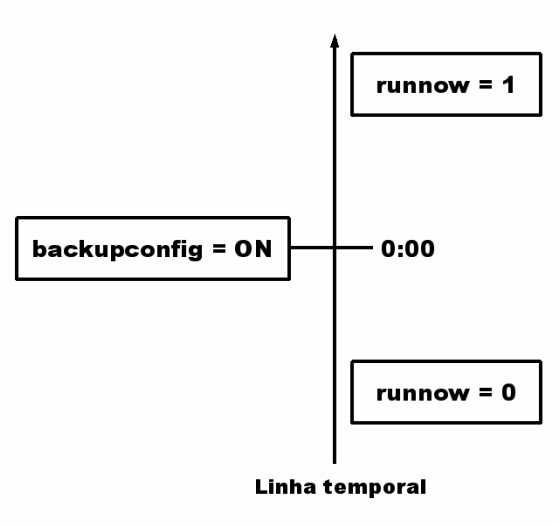



Figura 11 - Sistema automático de *backup* em andamento toda zero hora (0:00) para todos os dispositivos com a variável backupconfig marcadas como ON.

O motivo principal da realização constante de *backups* de arquivos de configuração de dispositivos no sistema está na necessidade de se ter uma cópia dos mesmos para cada equipamento monitorado. Todos os dados são armazenados em disco através do DBPortal. Esse é um procedimento comum em qualquer auditoria de *firewalls*. Em uma auditoria, a companhia sempre precisa conhecer o *status* temporal de todas as alterações nas configurações de seus dispositivos em monitoramento.

Ao ser acionado para realizar o *backup* de um dispositivo, duas situações distintas podem ocorrer durante o processo: o *backup* do dispositivo: ocorrer normalmente, ou acontecer uma falha no processo de *backup*. Se esse foi um *backup* bem sucedido, o sistema irá avaliar se o arquivo de configuração recebido é igual a gravação anteriormente registrada. Através do comando md5sum, o sistema tem como comparar os arquivos entre si – o já registrado no sistema e o recém adquirido pelo procedimento de *backup*.

É importante informar que, antes mesmo de executar quaisquer desses procedimentos de *backup*, a primeira ação tomada pelo sistema é a desativação da variável runnow do dispositivo em interação. Caso os arquivos sejam idênticos, o sistema irá descartar o arquivo mais recente e irá manter o arquivo mais antigo, que já está registrado em disco no DBPortal (veja a Figura

12). O sistema, então, irá registrar a data e a hora da ação na variável lastconfigbackup.

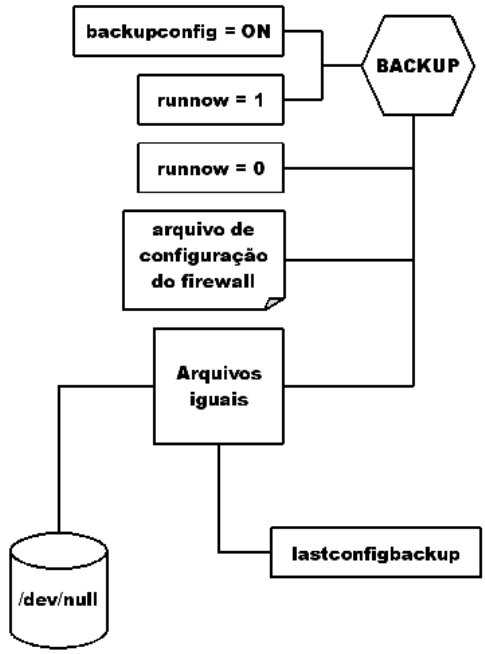



Figura 12 - Procedimento de *backup* onde os arquivos de configuração são idênticos.

Caso os arquivos sejam diferentes, será realizado um DIFF pelo sistema, onde as informações novas serão registradas em azul, e as informações descartadas, serão registradas em vermelho. O sistema então registra a nova configuração em disco. Após o procedimento, o DBPortal registra na variável lastbackupchange, a data e a hora do último *backup*. A data e a hora do processo também são registrados na variável lastconfigbackup (ver Figura 13).

Caso o processo de *backup* falhe, o sistema irá registrar a data e a hora do processo da tentativa de *backup*, tanto na variável lastconfigbackup quanto na lastbackupfailure (veja a Fig. 14). Para este dispositivo, mesmo com a falha no processo de realização de *backup*, o sistema só tentará efetuar um novo procedimento, caso ocorra uma intervenção manual do operador, sinalizando novamente a variável runnow do dispositivo para 1, ou de forma automática no

próximo ciclo de zero hora (0:00), caso o dispositivo ainda possua a sua variável backupconfig ativada.

Todos os dispositivos registrados no sistema possuem uma variável chamada backupfailures. Para todos, ela começa com o valor zero. A cada falha registrada no processo de *backup*, é adicionado 1 ao valor já encontrado nesta variável para esse dispositivo em questão. Esse valor só poderá ser resetado (i. é, voltar a ser zero) caso ocorra um procedimento de *backup* bem sucedido em uma próxima interação.

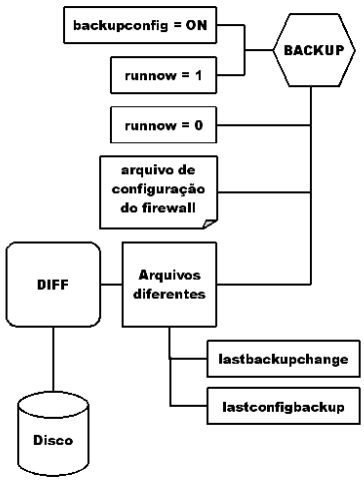



Figura 13 - Procedimento de *backup* onde os arquivos de configuração são diferentes.

Um dos problemas mais comuns para a ocorrência de uma falha no procedimento de *backup* é a falha na comunicação com o dispositivo. As formas mais comuns de ocorrência desse problema são: falha na resposta pelo dispositivo alvo, ou problemas de caminho da rede até o dispositivo. E essas situações são plenamente previstas pelo DBPortal. Veja a Figura 14 para maiores detalhes sobre o procedimento de *backup* quando ocorre uma falha.

Um dado importante a informar é que, independente do processo de *backup* falhar ou não, serão registradas a data e a hora da tentativa de gravação realizada pelo sistema. Essas informações sempre serão registradas no DBPortal, na variável lastconfigbackup. Para interromper os backups diários que se iniciam toda zero hora (0:00), os operadores precisam marcar

manualmente a variável backupconfig para OFF, para cada dispositivo onde a mesma se encontre ativa (ON).

Dentro do sistema DBPortal as variáveis backupconfig e runnow são consideradas variáveis booleanas no sistema de verificação de *backup*. Por segurança, as mesmas são representadas como colunas no banco de dados, dispensando qualquer interação flutuante, apenas em variáveis de memória do sistema em execução.

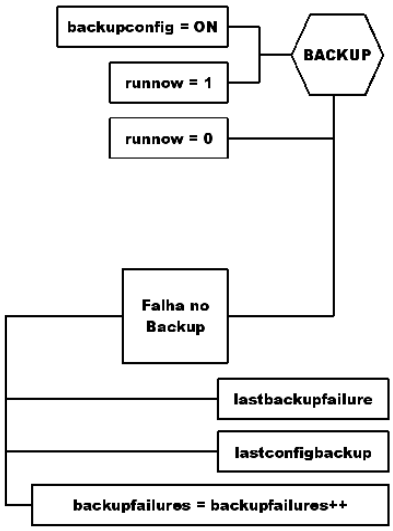



Figura 14 - Procedimento de *backup* não concluído.

O DBPortal é capaz de monitorar milhares de *firewalls* simultaneamente. O sistema permite mostrar todo o seu poder quando instalado para uso em grandes empresas. Nas grandes companhias, cada uma de suas unidades físicas (as mesmas, contendo milhares de *firewalls* cada) poderá estar espalhada por todo o planeta. Nesse tipo de cenário, a administração dos equipamentos no DBPortal é realizada por equipes de trabalhos distintas, separadas por área. Essa logística de operação permite que as equipes de trabalho criadas possam associar grupos de equipamentos para monitoramento local, otimizando assim todo o trabalho de auditoria em empresas desse porte.

A instalação do DBPortal é realizada em um único servidor, e as equipes cadastradas em todo o mundo acessam sua interface gráfica via *Web* para

realizarem suas tarefas. Como mencionado anteriormente, as únicas limitações do DBPortal são: sua capacidade de hardware, e a largura de banda associada para sua operação. Uma empresa de grande porte que queira utilizar o DBPortal para a automatização de seus processos de auditoria deve avaliar essas características físicas para poderem implementar o sistema de forma otimizada, atendendo assim as suas necessidades.

## H. Níveis de usuários

O DBPortal possui uma interface desenvolvida para uso com três níveis de usuários distintos: super-administrador, administrador e operador (Tabela 1):

1) Super-administrador: este usuário possui acesso completo as configurações do DBPprtal. O super-administrador é capaz de cadastrar equipamentos, criar grupos de trabalho por área de atuação (envolvendo um grupo regional de equipamentos a serem monitorados), pode criar novos administradores e operadores para o sistema, dentre diversas outras funções.
2) Administrador: o administrador possui poder de gerência local no grupo ao qual é designado para administrar, podendo cadastrar e descadastrar equipamentos para uso local em sua equipe. O mesmo é capaz de adicionar operadores para trabalhar no sistema em sua equipe.
3) Operador: O operador pode trabalhar com o monitoramento de equipamentos locais associados a sua equipe de trabalho. O mesmo é capaz de agendar *backups* para os equipamentos em monitoramento por sua equipe de trabalho.

Tabela 1: Quadro de usuários do DBPortal

| | super-administrador | administrator | Operador |
|---|---|---|---|
| Criação de usuários | SIM | SIM* | NÃO |
| Cadastro de equipamento | SIM | SIM | NÃO |
| Busca por equipamentos | SIM | SIM | SIM** |
| Monitoramento de equipes | SIM | SIM*** | NÃO |

* Capacidade apenas de criar usuários em nível de operador.
** Busca por equipamentos apenas em sua equipe de trabalho.
*** O monitoramento só pode ser realizado em sua equipe de trabalho.

É importante informar que, diante da relação de sigilo contratual existente entre a empresa e o desenvolvedor, o projeto apresentado não permite a divulgação de dados de qualquer natureza neste trabalho.

## I. Visão Geral do Funcionamento do DBPortal

O DBPortal possui grande parte dos recursos necessários para o trabalho diário de seus operadores, para a administração, monitoramento e auditoria de *firewalls*. Para facilitar a compreensão de seu funcionamento, será apresentada uma visão geral das principais funcionalidades do DBPortal e sua interface gráfica de operação. Após o operador acessar o DBPortal via login e senha, será carregada na mesma janela, a tela principal do sistema (Figura 15) evidenciando todos os recursos disponíveis para cada tipo de usuário (sendo mais específico, baseado na hierarquia de usuário). No exemplo abaixo a janela mostrará todas as funcionalidades do DBPortal disponíveis já que o *login* foi efetuado por um super-administrador do sistema.

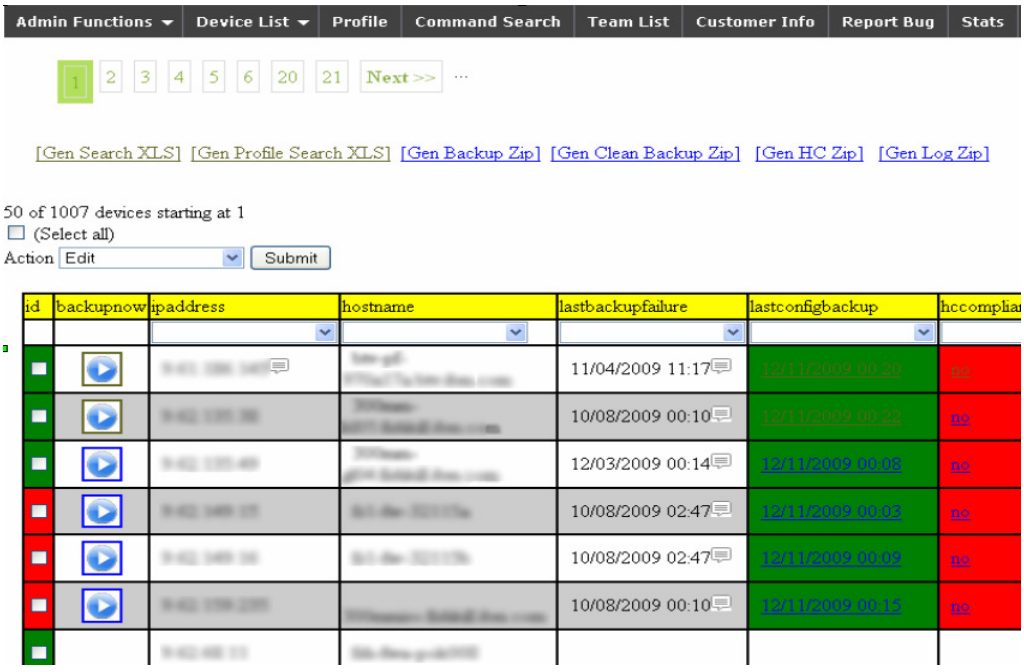



Figura 15 – Janela principal do DBPortal (algumas partes foram omitidas devido à necessidade de sigilo das informações)

No topo da janela encontramos o Menu Principal do DBPortal, com os itens *Admin Functions* (Funções do Administrador), *Device List* (Lista de Dispositivos), *Profile* (Perfil), *Command Search* (Busca por Comandos), *Team List* (Lista das Membros das Equipes, apresentando todas as informações referentes aos membros de cada equipe cadastrada), *Customer Info* (Informações sobre o Cliente), *Report Bug* (Relatório de Bugs) e *Stats* (Estatísticas). Abaixo do menu principal o operador encontra o sistema horizontal de navegação por páginas, onde a primeira página se encontra ativa e as demais navegáveis pelo operador. Por padrão, o DBPortal lista 30 (trinta) dispositivos cadastrados por página. Esse número pode ser facilmente alterado nas configurações de usuário de cada operador através do item de menu *Profile* do sistema. Logo abaixo encontra-se o segundo menu horizontal do DBPortal, com funções específicas de geração de conteúdo e arquivos de interesse para o sistema corrente de auditoria de cada empresa (será explicado em maiores detalhes mais abaixo). A seguir, identificamos a contagem total de dispositivos cadastrados no sistema, e quantos deles estão sendo apresentados na primeira página (*50 of 1007 devices starting at 1*).

Um dos recursos interessantes da interface do DBPortal é a caixa de marcação *Select All*, onde o operador do sistema poderá selecionar todos os dispositivos simultaneamente. Esse tipo de operação, não muito frequente, é uma funcionalidade extra para casos onde o operador precisa trabalhar com todos os dispositivos cadastrados de forma simultânea. Posteriormente, o DBPortal apresenta o seu menu de ação principal para os dispositivos cadastrados que estejam marcados para processamento (será explicado em maiores detalhes mais abaixo) e a listagem dos dispositivos em visualização, no formato de tabela. O sistema de apresentação dos dispositivos apresenta todas as ações e resultados cotidianos que cada operador precisa ter acesso sobre os dispositivos que esteja monitorando em sua equipe. A seleção de cada dispositivo para produção se encontra na coluna *id*. O DBPortal permite o processamento simultâneo de múltiplos dispositivos. Tudo o que o operador necessita é selecionar todos os dispositivos de interesse na coluna *id*. Na tabela de dispositivos destacamos as primeiras colunas:

1. *backupnow*, é apresentada como botões para cada dispositivo, onde o operador poderá ativar o *backup* do arquivo de configuração de cada dispositivo apresentado.
2. *ipaddress*, identifica o endereço de IP de cada dispositivo cadastrado no DBPortal.
3. *hostname*, identifica o nome de host de cada dispositivo cadastrado no DBPortal.
4. *lastbackupfailure*, informa a data e hora da última falha de backup para cada dispositivo.
5. *lastconfigbackup*, informa a data e a hora do último *backup* realizado do arquivo de configuração do dispositivo em questão.

O item *lastconfigbackup* para cada dispositivo é representado na forma de um link. Ao clicar nesse link, o operador poderá acessar o diretório respectivo desde dispositivo (Figura 16).

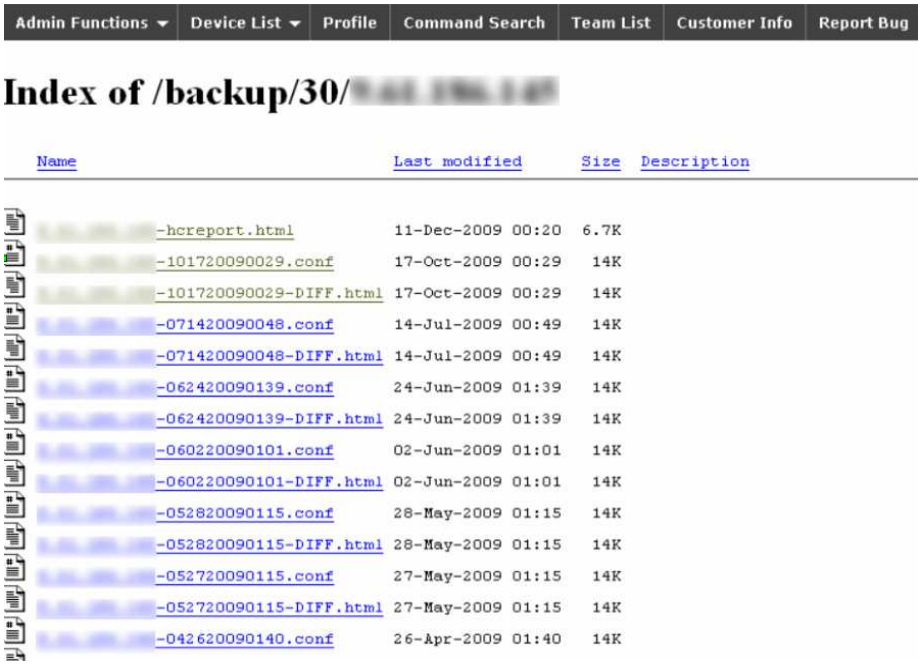



Figura 16 – Acesso ao diretório do dispositivo através do link apresentado para cada dispositivo na coluna *lastconfigbackup* do DBPortal *(algumas partes foram omitidas devido a necessidade de sigilo das informações).*

Nesta listagem de arquivos apresentada pelo diretório do dispositivo selecionado, destacamos os três primeiros. O primeiro arquivo é o HCREPORT (*healthchecking report*). Em formato HTML, esse é o arquivo de *healthchecking*

do dispositivo em questão (Figura 17). O segundo arquivo (*.conf*) é o último arquivo de configuração armazenado como *backup* para esse mesmo dispositivo (Figura 18). E o terceiro e último arquivo em destaque, é o arquivo que efetua o DIFF entre os arquivos de configuração dos dispositivos, caso os mesmos tenham sua configuração alterada ao longo do processamento via operador, durante os processos de *backup* realizados no sistema (Figura 19).

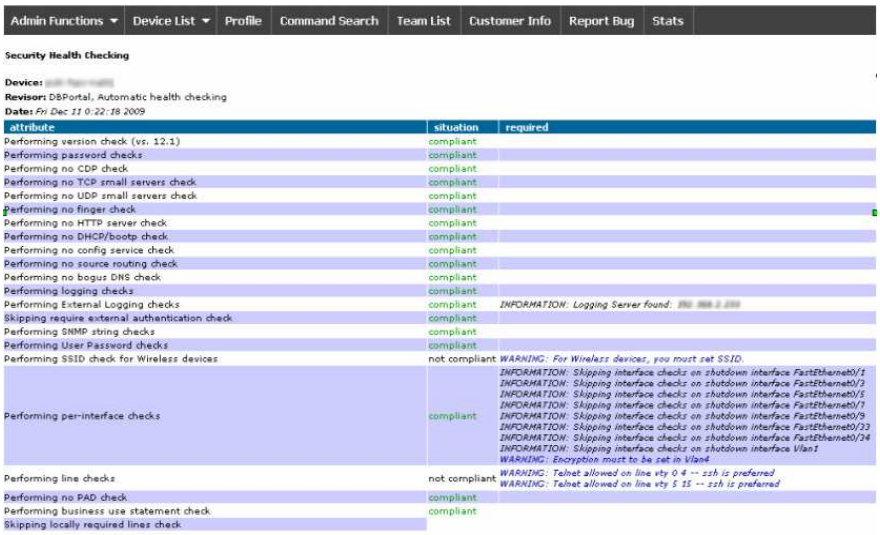



Figura 17 – Arquivo HCREPORT do dispositivo, apresentado em formato HTML o seu healthchecking (algumas partes foram omitidas devido a necessidade de sigilo das informações).

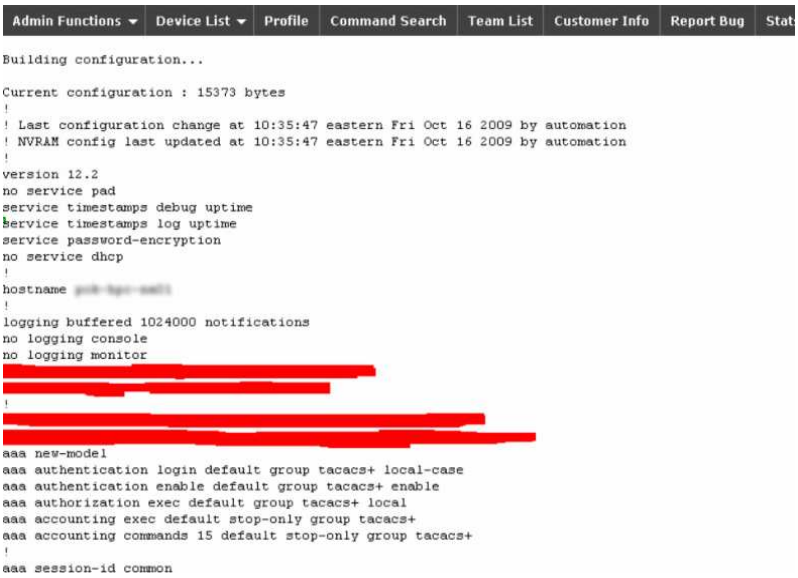



Figura 18 – Arquivo de configuração (formato texto) da última configuração válida do dispositivo cadastrado no DBPortal (algumas partes foram omitidas devido a necessidade de sigilo das informações).

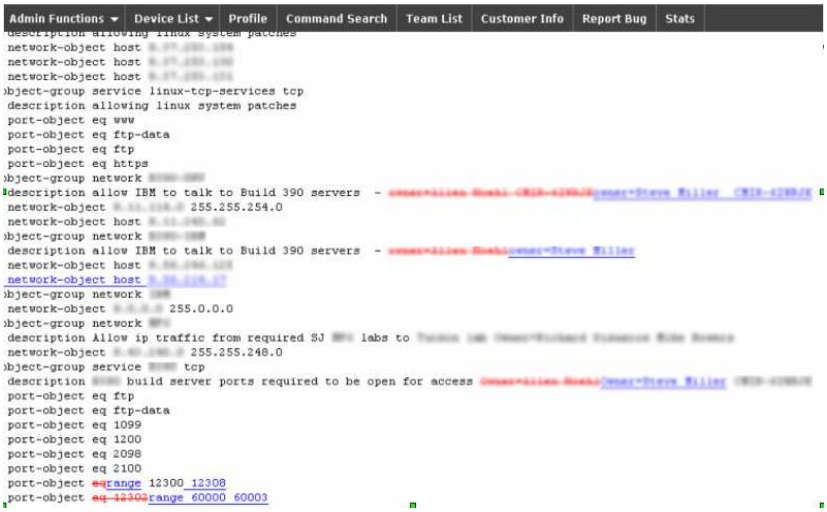



Figura 19 – Arquivo DIFF das configurações registradas em backup do dispositivo cadastrado no DBPortal (algumas partes foram omitidas devido a necessidade de sigilo das informações).

Para atender a todos os requisitos de uma auditoria destaca-se no DBPortal o cadastro do número de dias de *log* necessários para registro no sistema. O operador poderá processar e administrar esse recurso através da coluna *loggingdays* na tabela de dispositivos da tela principal do DBPortal (Figura 20). Como exemplo ilustrativo, o dispositivo registrado nesta figura possui a atividade de manter os seus últimos 05 (cinco) *logs* armazenados. Para acessar maiores informações sobre esses *logs*, o operador poderá clicar no balão de texto que se segue ao registro numérico de dias cadastrados para o armazenamento de *logs.*

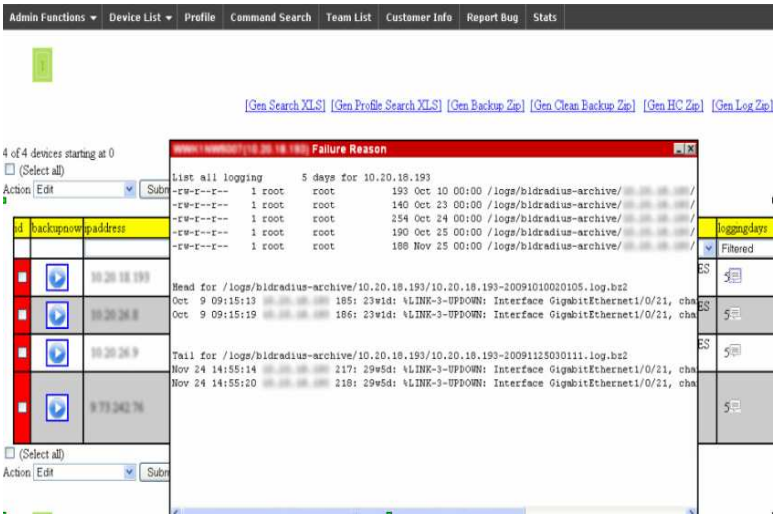



Figura 20 – A coluna loggingdays apresentando a caixa de registro expandida com a quantidade de dias de log para o dispositivo selecionado (algumas partes foram omitidas devido à necessidade de sigilo das informações).

Nesta nova janela, o operador poderá visualizar as informações derivadas da listagem dos últimos cinco *logs* desse dispositivo, o *head* (ou cabeçalho) do primeiro dos arquivos de *log*, e o *tail* (ou rodapé) do último arquivo de *log* do mesmo dispositivo. Esse é um padrão de apresentação de informações referentes à *logs* de dispositivos requeridos em uma auditoria de *firewalls*.

Outro item muito importante em uma auditoria é a exportação de informação em diversos formatos de arquivos. Formatos esses que estão a escolha de cada empresa. O atual DBPortal trabalha com a exportação de arquivos e informações em vários formatos diferentes, dependendo da tipo de informação especificada. Para auxiliar a automação dessa parte de um processo de auditoria extremamente necessária, foi criado o item de menu horizontal secundário. O primeiro item deste menu é o *Gen Search XLS*. O mesmo, efetua a exportação das informações dos dispositivos selecionados via coluna *id*, para o formato XLS da Microsoft™. O arquivo XLS gerado poderá ser utilizado na maioria das versões do aplicativo de planilha da Microsof™, o Excel® (Fig. 21).

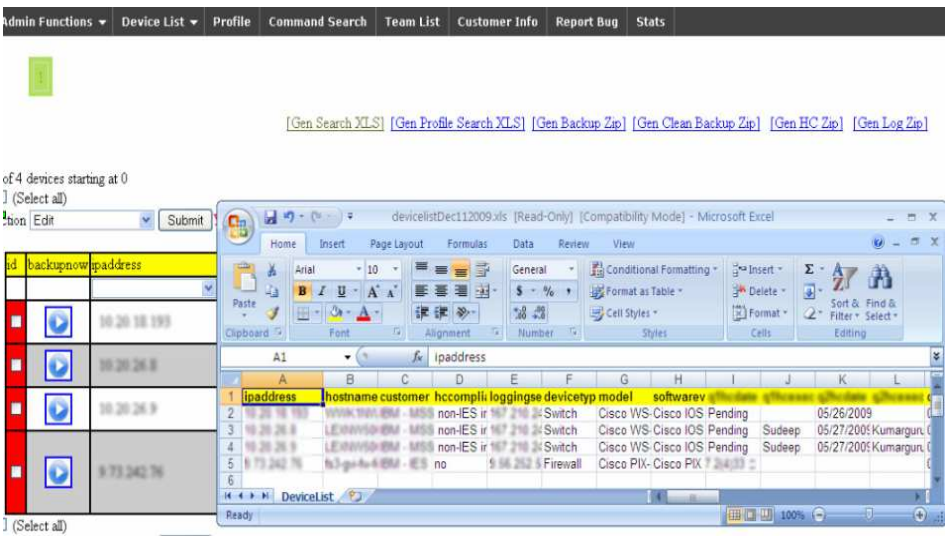



*Figura 21 – Uso do item de menu Gen Search XLS, permitindo exportar as informações sobre os dispositivos selecionados para um arquivo no formato XLS (algumas partes foram omitidas devido a necessidade de sigilo das informações).*

O exemplo apresentado, mostra a exportação e uso do arquivo XLS contendo todas as informações previamente listadas na tela principal do DBPortal, dos quatro primeiros dispositivos visualizados em tela pelo operador.

O segundo item de menu de importância (em análise) a ser citado é o *Gen Backup Zip*. Esse item é capaz de gerar um arquivo compactado no formato ZIP, dos últimos arquivos de backup para cada dispositivo selecionado (Figura 22).

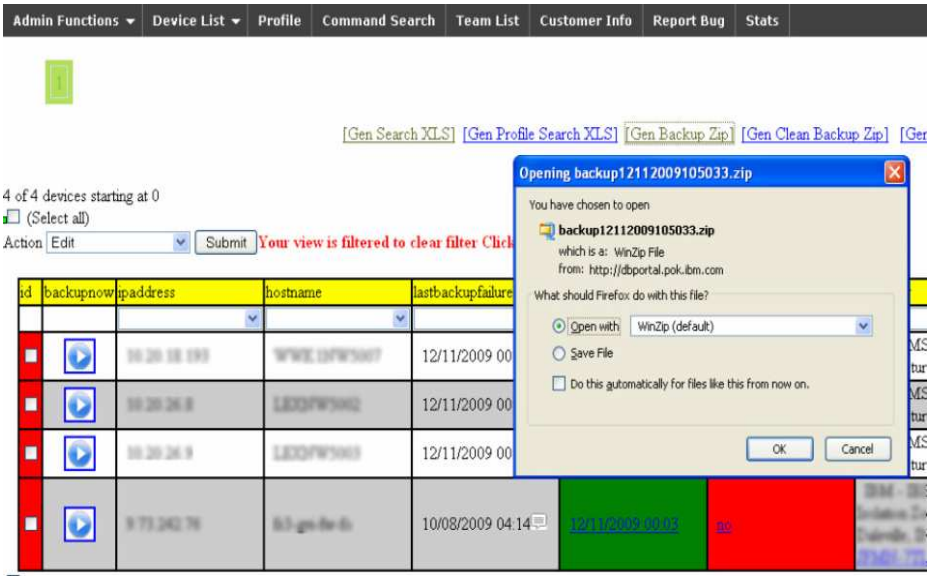



Figura 22 – Através do item de menu Gen Backup ZIP o DBPortal é capaz de gerar um arquivo compactado no formato ZIP, contendo o último arquivo de configuração de backup para cada dispositivo selecionado na tela de operação do sistema (algumas partes foram omitidas devido a necessidade de sigilo das informações).

O exemplo abaixo foi obtido através da seleção dos primeiros quatro dispositivos em tela. A figura mostra inclusive uma janela popup, sob o título *Opening backup12112009105033,zip*, para que o arquivo gerado possa ser salvo pelo operador do sistema.

Ao abrir o arquivo compactado no formato ZIP, é apresentado seu conteúdo constando de quatro arquivos de backup de configuração dos respectivos dispositivos selecionados (Figura 23). Inclusive é demonstrado a abertura de um dos arquivos de configuração e apresentado o mesmo em formato TXT via aplicativo NotePad® da Microsoft™. A seção distorcida da imagem refere-se as senhas desse dispositivo, que por motivos de sigilo, não podem ser apresentadas a público.

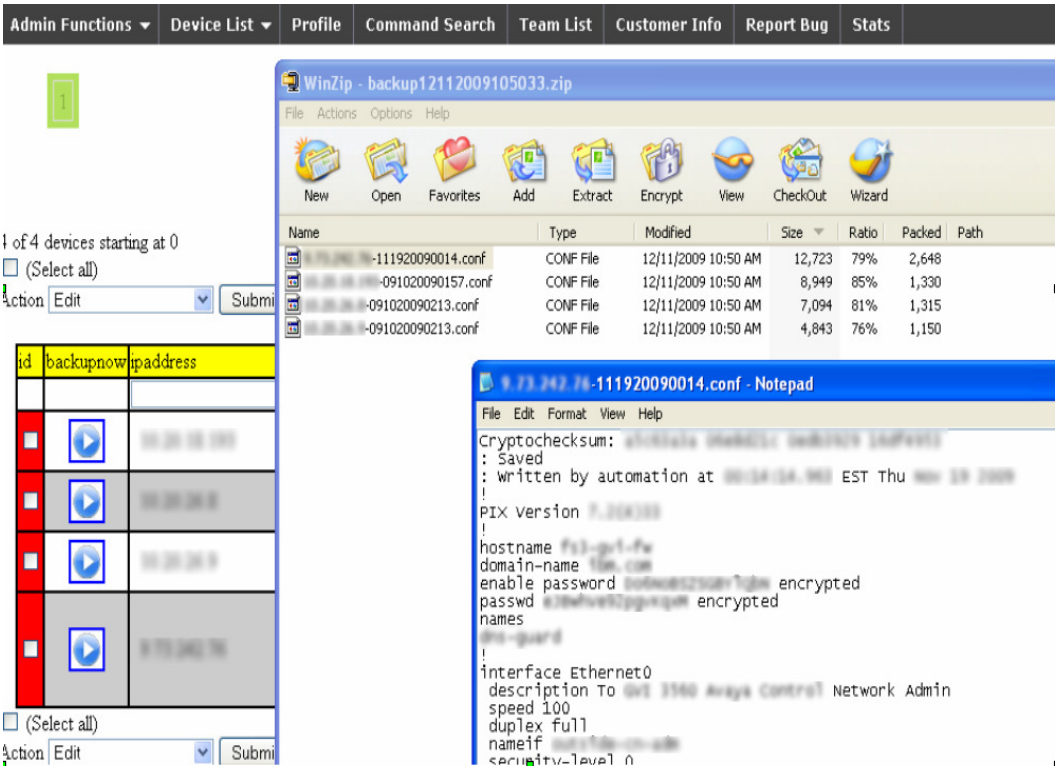



Figura 23 – Apresentação do conteúdo do arquivo compactado contendo os arquivos de configuração, gerados pelo Gen Backup ZIP, dos dispositivos selecionado. Inclusive é exibida a apresentação do conteúdo textual de um dos arquivos de configuração (algumas partes foram omitidas devido a necessidade de sigilo das informações).

A seguir é apresentado o item conjunto de ações do DBPortal através do menu de seleção vertical *Action* (Figura 24), localizado na tela principal da interface do sistema. O menu é composto das seguintes ações:

1) *Edit*, permite a edição das configurações de um dispositivo selecionado (edição individual para cada dispositivo selecionado).
2) *Bulk Action*, permite a edição de mais de um dispositivo selecionado de forma simultânea. É a versão de edição múltipla do item *Edit*.
3) *Copy*, permite a cópia de configuração entre os dispositivos cadastrados no DBPortal, facilitando o processo de administração dos dispositivos com configurações idênticas ou mesmo, semelhantes.
4) *Run Backup Now*, assim como presente na tabela de apresentação de dispositivos cadastrados, esse item permite a execução dos backups dos dispositivos selecionados. A vantagem desta ação é permitir a inicialização de backup de dispositivos simultâneos.
5) *Change Password*, permite a alteração da senha de usuário de acesso

(*access*) e de falha (*failsafe*) dos dispositivos cadastrados. A vantagem desse item no *Action* é a possibilidade de efetuar a alteração simultânea de senhas para todos os dispositivos cadastrados. Funciona como o sistema de *backup*, com o foco na troca de senhas dos dispositivos.

6) *Delete*, permite a deleção de um dispositivo do DBPortal, retirando-o da lista de dispositivos cadastrados no sistema.

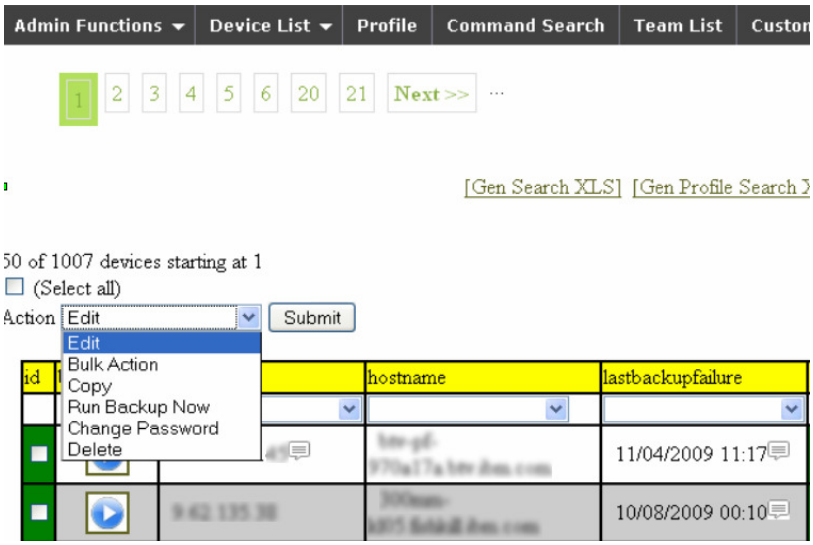



Figura 24 – Exposição dos itens de menu do Action, facilitando o processo de configuração global de dispositivos (algumas partes foram omitidas devido a necessidade de sigilo das informações).

No item *Edit* do menu *Action* o operador pode ter acesso a uma miríade de itens de configuração, todos de extrema importância para uso em auditoria de *firewalls* em grandes empresas (Figura 25). A primeira coluna de itens é a configuração global do dispositivo, onde pode-se ver a equipe responsável pelo equipamento, o IP do dispositivo, seu *host*, cliente, e diversos outros itens de igual importância. A título de sigilo de informação, algumas informações nesta figura foram camufladas.

A edição de configuração do dispositivo é extensa e separada por vários grupos de itens de configuração, como o *Healthchecking*, *Physical Location* (localização física), *Authentication* (Autenticação) e *Automation Settings* (Configurações de Automação).

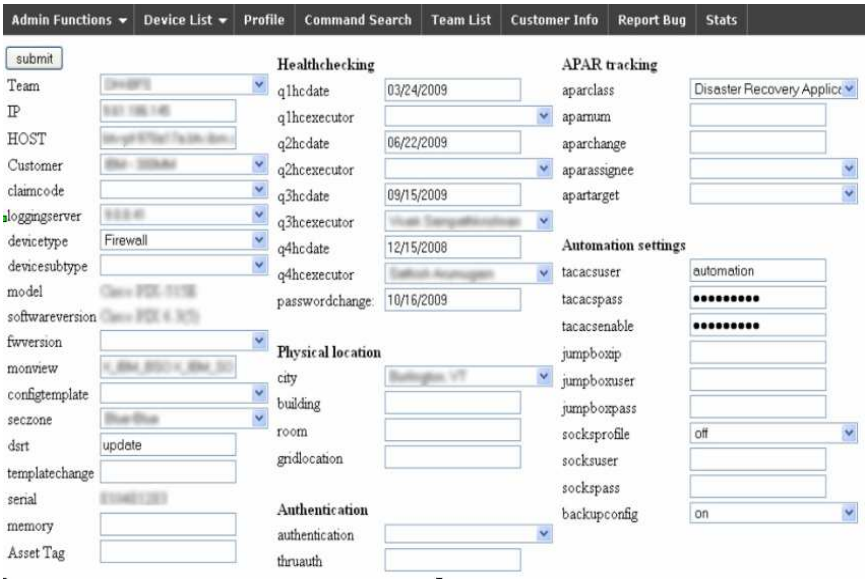



Figura 25 – Exemplo de edição de dispositivo cadastrado (ou a cadastrar) via item Edit do menu Action (algumas partes foram omitidas devido a necessidade de sigilo das informações).

Voltando ao menu principal do DBPortal, temos o item *Profile* (Figura 26). O Profile é onde o operador, independente de seu nível de usuário poderá configurar quais colunas quer ver (primeira coluna da esquerda para a direita) e qual a ordem em que elas aparecerão (segunda coluna). O que o operador estipular no *Profile* refletirá na tela principal do DBPortal. No Profile o operador também poderá atualizar e manter suas informações pessoais como nome, ID, telefone, dentre outros.

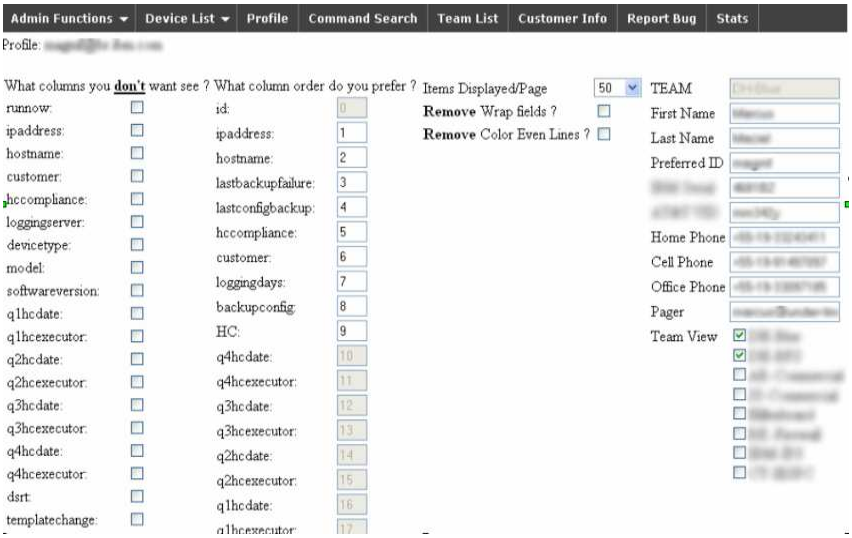



Figura 26 – Item de menu Profile sendo acessado, onde o operador poderá escolher e ordenar as colunas de apresentação dos dispositivos no DBPortal. Ele também poderá atualizar e manter os dados pessoais.

O operador também poderá escolher visualizar as outras equipes registradas no DBPortal, além da sua própria. Para trabalhos que envolvem mais de uma equipe esse recurso é essencial.

O item *Command Search* do menu principal do DBPortal é um dos recursos mais utilizados pelas equipes registradas no sistema (Figura 27). Através desta ferramenta o operador poderá efetuar buscas por comandos dentro dos últimos arquivos de configuração de todos os dispositivos registrados no sistema que estejam sob monitoramento por sua equipe. Essa funcionalidade facilita o trabalho dos operadores em encontrar dispositivos com configurações semelhantes dentro de seu universo de *firewalls* administrados, saber quais possuem determinados serviços habilitados, além de identificar suas vulnerabilidades.

No caso de vulnerabilidades, o operador poderá chegar ao dispositivo-alvo através da busca por comandos. Por exemplo: caso um dispositivo, de um modelo de fabricação e versão de sistema operacional específico possua uma vulnerabilidade na versão 1 do ssh, o operador poderá listar todos os dispositivos que utilizam o ssh versão 1, ao digitar no campo de busca "*ssh version 1*" e clicar no botão *Submit*. Dentre os resultados apresentados (Figura 28) o operador poderá localizar qual dispositivo que ainda utiliza a versão 1 do ssh de determinado modelo e versão de sistema operacional.




Figura 27 – Ferramenta Command Search do menu principal do DBPortal em atividade.

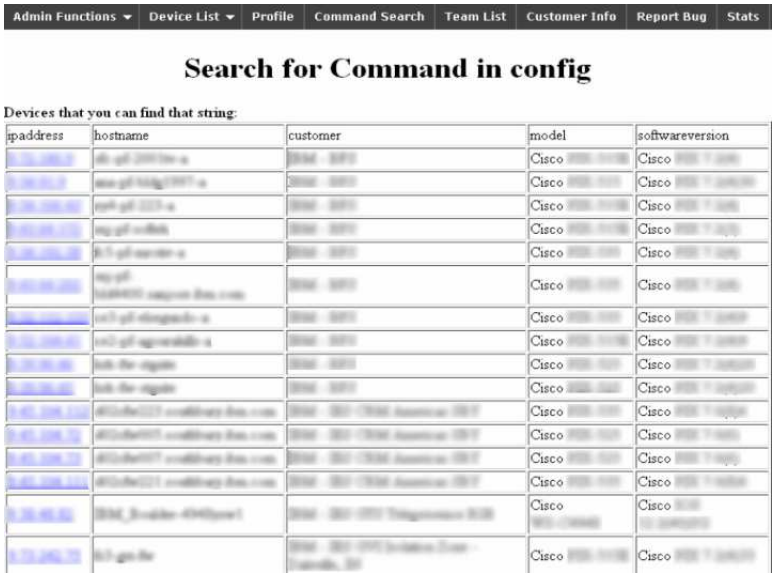



Figura 28 – Resultados da busca pela ferramenta Command Search.

## 6. Conclusão

O processo de auditoria de segurança em *firewalls* é um procedimento complexo e extenso, que certamente exige muito trabalho de toda uma equipe bem treinada. Este processo envolve a verificação periódica de todos os itens que fazem parte do procedimento de segurança da empresa. Com a utilização do sistema DBPortal estas atividades se tornam muito mais simples e rápidas, além de serem capazes de obter dados mais seguros e precisos, pois diminuem consideravelmente a possibilidade de erro humano em todo o processo. Através do uso do DBPortal a equipe responsável pelas auditorias é capaz de obter respostas em um curto intervalo de tempo, se comparado a um processo de intervenção manual.

O DBPortal, aparentemente, é uma ferramenta única no que se propõe. Suas funcionalidades não foram encontradas em nenhum outro software no mercado ou mesmo em projetos de software livre disponíveis ao público. Essa ferramenta pode ser utilizada para inventário de equipamentos de rede, monitoramento de alterações de modelo, versão, nº. Serial e configurações desses dispositivos, além de auxiliar as auditorias com um repositório de

informações afins. em uma rede empresarial. Porém seu foco atual é a auditoria de segurança de *firewalls* Cisco.

## Referências

[1] SCHMIDT, P.; SANTOS, J. L.; ARIMA, C. H. Fundamentos de auditoria de sistemas. São Paulo: Atlas, 2006.

[2] LYRA, M. R. Segurança e auditoria em sistemas de informação. Rio de Janeiro: Ciência Moderna, 2008.

[3] MITNICK, K. D.; SIMON, W. L. The art of deception: controlling the human element of security, ISBN: 85-346-1516-0, Pearson Makron Books, 2003.

[4] COOK, B. Fortigate-60 Firewall Security Audit: An Auditor's Perspective, https://it-audit.sans.org/community/papers/fortigate-60_firewall_security_audit:_an_auditors_perspective_178, Acessado em 16 de nov. 2009.

[5] YUEN, R. W. Auditing a Cisco PIX firewall: An Auditor Perspective, https://it-audit.sans.org/community/papers/auditing_a_cisco_pix_firewall:_an_auditor_perspective_70, Acessado em 16 de nov. 2009.