UNIVERSIDADE FEDERAL DO ABC

DIVISÃO ACADÊMICA DO CECS

TRABALHO DE GRADUAÇÃO III

APACHE SPOT – BIG DATA E SEGURANÇA DA INFORMAÇÃO

GUILHERME BUENO AOKI – RA: 11132209

SANTO ANDRÉ

ABRIL DE 2018

UNIVERSIDADE FEDERAL DO ABC

Engenharia da Informação

TRABALHO DE GRADUAÇÃO III

Aluno: Guilherme Bueno Aoki

Orientador: Prof. Dr. Ricardo Suyama

Trabalho de Graduação III apresentado ao Curso de Graduação em Engenharia da Informação como requisito parcial para obtenção do grau de Engenheiro da Informação

SANTO ANDRÉ

ABRIL DE 2018

FOLHA DE APROVAÇÃO

GUILHERME BUENO AOKI

CONCEITO FINAL

_____

ORIENTADOR:

Ricardo Suyama

Assinatura: _____

RESUMO

O objetivo deste trabalho foi realizar a implantação da ferramenta Open Network Insight, rebatizada como Apache Spot, num ambiente de testes controlado, com fluxos e pacotes de dados disponibilizados na Internet. Com este trabalho, foi possível observar todas as funcionalidades da ferramenta, num escopo reduzido de testes, para avaliar seu desempenho e funcionalidades como ferramenta na área de Segurança da Informação. Desenvolvida recentemente (ano 2016), foi aberta para contribuição da comunidade através do projeto Apache e acredita-se ter um forte potencial no mundo da Tecnologia da Informação. Neste trabalho, foi considerado um conjunto de dados simulando 6 milhões de requisições DNS no período de uma hora de captura de dados. Dentro desses dados, foram inclusos pacotes artificiais com um ataque de tunelamento em simples troca de requisição/resposta DNS utilizando TCP sobre o DNS, apresentando uma carga codificada na requisição DNS utilizando hex0x20Hack. Para análise, foi necessária a instalação de um ambiente Hadoop e da ferramenta Apache Spot, ingestão dos dados mencionados e sua análise na interface gráfica do Apache Spot, com o objetivo de avaliar seus potenciais benefícios.

ABSTRACT

The main objective of this work is to install a tool called Open Network Insight, now known as Apache Spot, in a controlled tests environment with data flows shared on the Internet. With this work, we'll be able to observe all the tool's features in a minimal way, proving it's value in the Information Security Market. Developed recently (2016), it's code was opened for contribution through the Apache project and it is believed to have a powerful potential in the IT World. In this work, it was considered a dataset simulating 6 million DNS queries within one hour of data capturing. Inside this dataset, was included artificial packets emulating tunneling simple HTTP Request/Response exchange using TCP over DNS, presenting an encoded payload in the DNS requests by using hex0x20Hack. For the analysis, it was necessary the Hadoop environment and Apache Spot installation, ingestion of the mentioned dataset and analysis through Apache Spot's graphic interface, meaning to evaluate its potential benefits.

SUMÁRIO

1 INTRODUÇÃO

Ao mesmo tempo que as redes espalhadas pelo mundo crescem exponencialmente, elas se tornam mais complexas e de missão crítica (como por exemplo, diversos sensores conectados numa aeronave, ou controladores de grandes máquinas numa indústria e até o a rede do sistema metroviário), trazendo novos desafios para os responsáveis por gerenciá-las. A necessidade por uma infraestrutura de rede integrada que compreende serviços de voz, vídeo e dados (all-in-one) é evidente, mas esse crescimento rápido das tecnologias introduz preocupações novas com a segurança.

Sem as devidas garantias, todas as partes de uma rede são vulneráveis a uma falha de segurança ou atividade não-autorizada de intrusos, competidores ou até funcionários da empresa. Várias das organizações que gerenciam sua própria segurança nas redes internas e usam a Internet com mais objetivos do que receber e enviar e-mails costumam ter experiências com ataques à sua rede – mais da metade dessas empresas não sabem que estão sendo atacadas. Empresas menores normalmente estão satisfeitas, por ter ganhado um falso senso de segurança. Eles usualmente reagem ao último vírus ou ataque ao site, mas estão presos numa situação onde não possuem os recursos e tempo necessário para colocar em segurança.[STALLINGS W. 2011]

Dessa forma, o desafio de manter sua infraestrutura de redes segura nunca foi tão grande ou tão crucial para os negócios. Apesar dos consideráveis investimentos em segurança da informação, as organizações continuam a ser atingidas por incidentes cibernéticos e ao mesmo tempo, tentam gerenciar seus focos para melhores resultados com menos recursos. Consequentemente, melhorar a área segurança se mantém essencial para garantir o funcionamento dos sistemas sem diminuir a flexibilidade da companhia.

Embora seja possível encontrar na literatura inúmeras orientações sobre quais aspectos devem ser analisados para orientar o planejamento, design e implantação de uma rede segura, essencialmente é necessário ter respostas claras para algumas questões fundamentais, como [BHAJI Y. – 2008]:

- O que será protegido ou gerenciado?
- Quais são os objetivos do negócio?
- O que é necessário para esses objetivos?

- Qual tecnologias ou soluções são necessárias para alcançar esses objetivos?
- Estes objetivos são compatíveis com a segurança de infraestrutura, operações e ferramentas?
- Quais riscos são associados com uma segurança inadequada?
- Quais são as implicações de não implantar segurança?
- Serão introduzidos novos riscos não cobertos por suas atuais políticas de segurança?
- Como é possível reduzir os riscos?
- Qual é a tolerância para os riscos?

Estas perguntas podem ser utilizadas para estabelecer uma postura e definir prioridades sobre requisitos fundamentais para uma rede segura.

1.1 TRANSFORMAÇÃO DO PARADIGMA DE SEGURANÇA

Com o aumento do tamanho das redes, os ataques se tornam cada vez mais sofisticados e o pensamento sobre segurança vem se alterando. Alguns dos principais fatores que têm contribuído para a mudança nos paradigmas de segurança podem ser resumidos em quatro pontos [NAKAMURA E., P. GEUS – 2010]:

- **Segurança não é mais sobre "Produtos"**: Soluções de segurança precisam ser escolhidas com os objetivos de negócio em mente e integrados com as ferramentas e procedimentos operacionais;
- **Demandas em escala estão aumentando:** Com o aumento de vulnerabilidades e ameaças de segurança, as soluções precisam escalar para milhares de servidores em grandes empresas;
- **Legados de segurança e custos são um desafio:** Produtos reativos forçam a implantação e renovação de múltiplos agentes e gerenciamentos de paradigmas;
- **Dano do Dia Zero:** Ataques com rápida propagação (Slammer, Numda, MyDoom) acontecem de forma muito ágil para produtos reativos controlarem. Portanto, um autônomo e proativo sistema de segurança é necessário para combater os vírus e worms extremamente dinâmicos.

Com as redes distribuídas atuais, a segurança não pode ser reforçada apenas nas bordas ou perímetro. Ataques do Dia Zero (ou novos vírus introduzidos pela primeira vez)

continuam a infectar empresas e serviços de provisão de redes. A solução usual das empresas, a fim de estabilizar a proteção contra tais ataques, tem sido desenvolver pacotes para os sistemas a cada vez que as vulnerabilidades se tornam conhecidas. Este movimento, entretanto, não pode escalar em grandes redes e essa situação apenas pode ser endereçada com sistemas proativos em tempo real.

Por esses motivos, a segurança hoje tem sido principalmente focada no gerenciamento e redução de riscos em ambientes com rápido desenvolvimento. A redução de riscos máxima é encontrada com uma solução integrada construída sobre uma infraestrutura inteligente e flexível, com operações e ferramentas eficazes, alinhada aos objetivos da área de negócios.

1.1 PRINCÍPIOS DE SEGURANÇA

Um modelo de segurança simples, mas amplamente aplicado, é a tríade Confidencialidade, integridade e disponibilidade [BHAJI Y. – 2008], conforme ilustrado na Figura 01. Estes três princípios chave devem guiar todos os sistemas de segurança. Eles também proveem uma ferramenta para mensurar implantações de segurança. Estes princípios são aplicáveis por todo o espectro de análise de segurança – desde o acesso até o histórico de acesso à Internet do usuário, até a segurança de dados criptografados na Internet. Uma brecha em qualquer um destes princípios pode gerar sérias consequências para todas as partes envolvidas.

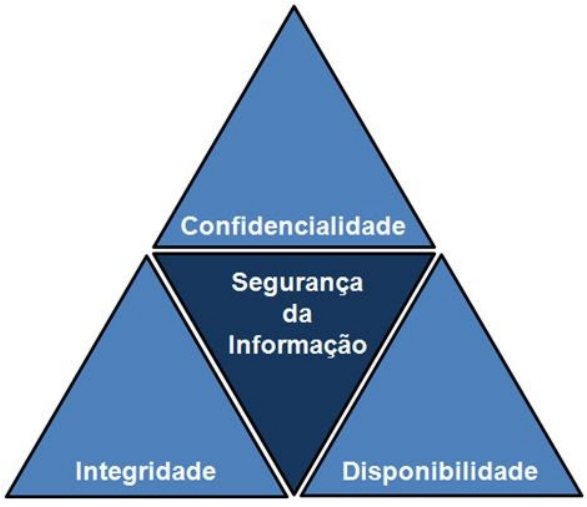


Figura 01: Tríade Confidencialidade, Integridade e Disponibilidade

1.1.1 CONFIDENCIALIDADE

A confidencialidade previne a divulgação não autorizada de informações sensíveis. É a capacidade de garantir que o nível de sigilo seja reforçado e a informação não seja revelado para usuários não autorizados. Tratando-se de segurança, a confidencialidade provavelmente é o aspecto mais óbvio de todos e é o que costuma ser mais atacado. Métodos de criptografia são exemplos de tentativas de garantir a confidencialidade dos dados transferidos entre um computador e outro. Por exemplo, durante transações bancárias online, o usuário quer proteger a privacidade dos detalhes da conta, como senhas e números de cartão. A criptografia consegue prover uma transmissão segura, protegendo os dados sensíveis.

1.1.2 INTEGRIDADE

A integridade previne modificações não autorizadas de dados, sistemas ou informações, de modo a prover segurança da acurácia das informações e sistemas. Se os dados são íntegros, é possível ter certeza que são precisos e não foram modificados da fonte original de informações. Um tipo comum de ataque de segurança é o “Man-In-The-Middle”. Este ataque, o intruso intercepta os dados durante a transferência e os modifica, fazendo com que o receptor não receba as informações corretamente.

1.1.3 DISPONIBILIDADE

A disponibilidade é a prevenção de perdas de acesso aos recursos e informações, garantindo que os sistemas estejam disponíveis quando necessário. É imperativo garantir que as informações estejam prontamente acessíveis para os usuários autorizados durante todo o tempo. “Denial of Service” (DoS) é um dos tipos de ataques de segurança mais severos e tentam negar o acesso para os usuários com a disrupção dos serviços.

1.2 POLÍTICAS, PADRÕES, PROCEDIMENTOS, LINHAS DE BASE E GUIAS

Um modelo de segurança é uma estrutura de múltiplas camadas, criada com várias entidades integradas e mecanismos de proteção lógicos e físicos, todos trabalhando em conjunto para prover um sistema seguro que cumpre as melhores práticas.

1.2.1 POLÍTICAS DE SEGURANÇA

Uma política de segurança é um conjunto de regras, práticas e procedimentos que ditam como informações sensíveis são gerenciadas, protegidas e distribuídas. No mundo da segurança de redes, as políticas normalmente são pontuais, ou seja, cobrem apenas uma única área. Uma política de segurança é um documento que expressa exatamente o nível de segurança baseado nas metas que os mecanismos de segurança devem atingir. Deve ser escrita por uma equipe de alta gerência para descrever o que é a segurança da informação.

Confiança é um dos principais temas em muitas políticas. Algumas empresas não possuem uma política porque elas confiam em seu corpo de funcionários e confiam que todos serão corretos. Entretanto, nem sempre este é o caso. A maioria das organizações precisa de políticas para garantir que todos são regidos pelas mesmas regras.

As políticas tendem a aumentas a apreensão das pessoas, pois elas não querem estar presas a regras e regulamentos. Uma política deve definir o nível de controle de usuários e precisa observar e balancear com as metas de produtividade. Uma política muito restrita será difícil de implantar, pois suas regras serão minimamente cumpridas ou ignoradas. No outro extremo, uma política muito leve pode ser evadida e não garante a responsabilidade dos usuários e rastreamento do ponto de vulnerabilidade (pessoa ou equipe envolvida). É necessário que seja balanceada de acordo com os objetivos da empresa.

Dependendo do tamanho da organização, diversos tópicos das políticas de segurança são potencialmente apropriados. Para algumas organizações, um grande documento cobrirá todas as facetas; para outras, vários pequenos documentos individuais e focados são necessários. São alguns exemplos de políticas comuns que uma organização pode considerar [BHAJI Y. – 2008].:

- **Uso aceitável:** Esta política contorna o uso aceitável de computadores. As regras são estabelecidas para proteger o funcionário e a empresa. O uso inapropriado expõe a companhia a riscos, incluindo ataques de vírus, comprometimento de sistemas e serviços de rede e problemas legais.
- **Ética:** Esta política põe ênfase nas expectativas do funcionário e do consumidor em cumprir as práticas de regras justas ao negócio. Estabelece uma cultura de abertura,

confiança e integridade nas práticas de trabalho. Esta política pode guiar o comportamento do negócio para garantir a conduta ética.

- **Sensibilidade das informações:** Esta política tem a intenção de ajudar os funcionários a determinar que tipo de informação pode ser aberta ao mundo externo, tanto como determinar qual tipo de informação não pode ser transmitira sem a devida autorização. A informação coberta nessas diretrizes inclui todos os tipos de informação eletrônica, em papel, divulgada oralmente ou visualmente.
- **E-mail:** Esta política cobre o uso apropriado de qualquer e-mail enviado por um endereço da organização e é aplicado a todos os seus funcionários, fornecedores e agentes que trabalham de alguma forma para a companhia.
- **Senhas:** O propósito desta política é estabelecer um padrão de criação de senhas fortes, sua proteção e uma frequência de mudança.
- **Avaliação de Riscos:** Esta política é usada para reforçar que o grupo de Segurança da Informação faça avaliações de risco periódicos com o propósito de determinar áreas de vulnerabilidade e iniciar os reparos apropriados.

1.2.2 PADRÕES DE SEGURANÇA

Padrões são as melhores práticas de segurança reconhecidas pela indústria. Através de padrões, estruturas e princípios são acordados e modelados para implantar e manter o nível de segurança necessário nos processos de uma companhia.

Assim como políticas de segurança, padrões são de natureza estratégica e definem parâmetros de sistemas e procedimentos. Eles variam por indústria na qual são aplicadas.

Como por exemplo, um padrão amplamente difundido no ambiente de Segurança da Informação é a norma ISO 27001, que adota um processo de gestão, considerando uma série de requisitos, processos e controles com o objetivo de mitigarem os riscos da organização. Este é um padrão que existem empresas certificadoras para as boas práticas referentes à gestão de segurança, com todo o planejamento, liderança, suporte, operação, avaliação e melhoria dos ambientes.

Outro padrão adotado é a ISO 27002, que adota as melhores práticas na área da segurança da informação, assim como a 27001. [International Organization for Standardization]

Padrões gerais também são determinados pelo COBIT (Control Objectives for Information and Related Technologies) e pelo ITIL (Information Technology Infrastructure Library), mas englobando todo o mundo da tecnologia da informação e podendo ser aplicados como boas práticas para quaisquer tipos de empresas para a operação e gerenciamento de serviços.

1.2.3 PROCEDIMENTOS DE SEGURANÇA

Procedimentos são documentos de "baixo nível" que definem instruções sistemáticas em como as políticas de segurança e padrões são implantados num sistema. Procedimentos são detalhados para prover o máximo de informação aos usuários para que eles possam implantar e reforçar com sucesso as políticas de segurança e aplicar os padrões e diretrizes de um programa de segurança.

Funcionários usualmente se referem a procedimentos mais frequentemente do que outras políticas e padrões, pois os procedimentos revelam os detalhes atuais de uma fase de implantação de um programa de segurança.

Diversas companhias possuem procedimentos de segurança para proteger suas informações. São criados com o objetivo de manter todo o ambiente seguro de acordo com as auditorias que são feitas periodicamente ("compliance") e baseadas em diversos tipos de normativas, como descrito anteriormente. [NAKAMURA E., GEUS, P. – 2010]

Diversos exemplos de procedimentos de segurança podem ser implantados em companhias. Entretanto, são descritos de acordo com a sensibilidade das informações que a empresa trafega, os critérios de seus clientes (ou áreas clientes) e a necessidade de atingir patamares elevados ou não de segurança. Alguns procedimentos de segurança podem ser:

- Restrição de acesso à ambientes físicos com autenticação biométrica;
- Utilização de uma base de usuários com suas determinadas autorizações para acesso à informação;
- Criação de um documento padrão de normatização para instalação de software, com o objetivo de garantir que as políticas e padrões de segurança sejam seguidos;
- Procedimentos de liberação de regras de Firewall para comunicação entre servidores e usuários;

- Saneamento de usuários que estão fora de utilização após determinado período;
- Procedimentos de penetração em ambientes com o objetivo de validar se todas as novas vulnerabilidades lançadas (por mudança de versão de Sistema Operacional, por exemplo) foram sanadas;
- Procedimentos de aprovações para criação e autorização de usuários nos sistemas;

1.2.4 LINHAS DE BASE DE SEGURANÇA

Uma linha de base é o nível mínimo de segurança necessário num sistema. Linhas de base podem prover aos usuários, meios para alcançar o nível mínimo absoluto de segurança necessário que é consistente com todos os sistemas da organização.

Como por exemplo, usuários que acessam bancos de dados, não devem poder ter privilégios para escrita e exclusão de dados, desta forma podem evitar uma perda de informações extremamente importantes através de funcionários tentando prejudicar a empresa.

Ao que se diz respeito à infraestrutura, pode ser realizado uma checagem com diversos pontos para garantir esta linha de base, como por exemplo (CISCO – 2008):

- Revisar todas as portas e serviços terminais e de gerência disponíveis;
- Desabilitar todas as portas que não são explicitamente necessárias ou que estão em utilização;
- Autorização de apenas acessos à portas e serviços liberados para as origens que estão autorizadas;
- Negar acesso ao mundo externo, apenas se for explicitamente requisitado;
- Detectar e fechar seções que estão inativas;
- Restringir o número de acessos concorrentes vindo de um mesmo destino;

Ações como esses exemplos listadas poderão auxiliar as ferramentas de segurança para garantir a segurança de sistemas corporativos.

1.2.5 DIRETRIZES DE SEGURANÇA

Diretrizes de segurança são as ações recomendadas e as operações necessárias para os usuários. Similarmente aos procedimentos, as diretrizes são táticas em sua natureza. A

maior diferença entre padrões e diretrizes é que as diretrizes podem ser usadas como referência, enquanto padrões são ações mandatórias na maior parte dos casos.

De acordo com o tipo de negócio, as diretrizes podem mudar. Empresas extremamente complexas e grandes, devem aumentar a governança de seus dados e acessos, de forma a garantir que todo o acesso seja controlado. Empresas menores, com maior agilidade, podem possuir diretrizes menos fortes com relação à segurança, devido ao seu maior controle sob seus funcionários.

1.3 MODELOS DE SEGURANÇA

Um importante elemento no desenho e análise de sistemas de segurança é o modelo. Isto acontece pois ele integra as políticas de segurança que devem ser reforçadas no determinado sistema. Um modelo de segurança é o retrato simbólico de uma política de segurança. Ele mapeia os requisitos dos legisladores num conjunto de regras e regulamentos que precisam ser seguidos por um sistema de computadores um de redes. Uma política de segurança é um conjunto de metas abstratas e requisitos de alto nível e o modelo sãos as ações necessárias e o que não deve ser feito para que as metas sejam alcançadas. De forma mais simplificada, temos os seguintes modelos [NAKAMURA E., GEUS, P. – 2010]:

- **Modelo Bell-LaPadula (BLM),** também chamado de modelo multi-nível. Foi introduzido principalmente para reforçar o controle de acesso para aplicações governamentais e militares. O BLM protege a confidencialidade da informação dentro de um sistema (BELL, D. - 2010);
- **Modelo Biba** é uma modificação do BLM que principalmente reforça a integridade da informação dentro de um sistema (BALON, N. - 2004);
- **Modelo Clark-Wilson:** Previne usuários autorizados de realizar modificações não-autorizadas nos dados. Este modelo introduz um sistema de trios: um sujeito, um programa e um objeto (BLAKE, S. - 2000);
- **Modelo de Fluxo de Informações:** Restringe os fluxos de informações para que apenas se mova entre níveis de segurança aprovados (MCLEAN, J. - 1990);

- **Modelo Muralha da China:** Combina a discrição comercial com controles mandatórios obrigatórios. É necessário para as operações financeiras de muitas organizações (GUPTA, V. - 2009);
- **Modelo Lattice:** Lida com informações militares. Controles de acesso baseado neste modelo foram desenvolvidos no início dos anos 70 para lidar com a confidencialidade de informações militares. Ao fim da década de 70, pesquisadores aplicaram este modelo a certas preocupações de integridade. Posteriormente, aplicações de modelos à política da Muralha da China, uma política única para o setor comercial foi desenvolvida e uma perspectiva balanceada no controle de acesso baseado no modelo Lattice foi criada, desta forma (SANDHU, R. - 1993);

Para todos os modelos citados anteriormente, o Apache Spot pode ser utilizado de forma a monitorar todas as atividades dentro do ambiente de forma rápida e assertiva. O objetivo da ferramenta, neste momento, é apenas facilitar a monitoração e a ingestão de diversos tipos de arquivos diferentes para a análise. No futuro, talvez ela poderá ser integrada e melhorada para tomar ações automaticamente, restringindo acessos não autorizados e prevendo ataques futuros aos ambientes monitorados.

As suas características reforçarão qualquer um dos modelos apresentados, garantindo que seja monitorado de acordo com todas as regras de negócio já impostas neles e alertando aos analistas de segurança, e também de forma preventiva, possíveis ataques e acessos não autorizados.

1.4 PERÍMETRO DE SEGURANÇA

As opiniões a respeito do perímetro de segurança mudaram nos últimos anos. Parte desta mudança é que a natureza do perímetro de segurança se tornou cada vez mais incerta e cada pessoa possui uma visão diferença dele. Os limites do perímetro se tornaram extensos, sem ligações geográficas e o acesso remoto se tornou parte integrante da rede [NAKAMURA E., GEUS, P. – 2010].

Na essência, o perímetro se transformou e estendeu para diversos níveis dentro de uma rede. Em outras palavras, as redes hoje em dia não possuem apenas uma entrada. Elas são acessíveis de diversas formas em ambientes abertos e precisam de um controle de acesso para toda a rede. A transformação leva a um pensamento de redes de multi-perímetro.

Com o aumento das redes tradicionais e os acessos remotos de redes sem fio, notebooks, celulares, e inúmeros outros dispositivos, medidas são necessárias para suprir estas redes. O conceito de “dentro versus fora” se torna muito complexo. Por exemplo, quando se conecta em uma rede corporativa utilizando uma rede privada virtual (VPN), não mais se está “fora” da rede corporativa. Todos os serviços em execução no computador estão “dentro” da rede.

Negócios com redes globais confiam em suas redes para se comunicar com seus funcionários, clientes, parceiros e fornecedores. Apesar de que acesso imediato às informações e comunicações são uma vantagem, traz preocupações sobre segurança e proteção de acesso à recursos de rede críticos.

Administradores de rede precisam saber quem acessa, quais recursos e estabelecer claros perímetros para controle de acesso. Uma política de segurança efetiva balanceia a acessibilidade com a proteção. Políticas de segurança são reforçadas nos perímetros da rede. Frequentemente existe a concepção de um perímetro como a fronteira entre a rede interna e a Internet pública. Entretanto, um perímetro pode ser estabelecido em qualquer lugar dentro de uma rede privada ou entre sua rede e a rede de um parceiro.

Um perímetro compreensível para uma solução de segurança abre as comunicações conforme definido pela política de segurança, protegendo os recursos da rede de brechas, ataques ou uso não autorizado e controla múltiplos pontos de acesso e saída da rede. Também aumenta a garantia do usuário com múltiplas camadas de segurança.

O Apache Spot trabalha de acordo com a figura 02, no que diz respeito ao perímetro de segurança:

Figura 02: Perímetro de Segurança Apache Spot [Site oficial Apache Spot – 2018]

De acordo com a figura 02 acima, a ferramenta é capaz de trazer uma série de caraterísticas capazes de abordar os mais diversos tipos de dados (Fluxo de dados internos e externos, análise de pacotes DNS e Proxy). Desta forma, se tornaria uma ferramenta extremamente completa em termos de monitoramento de todo o perímetro de segurança de um Data Center.

1.5 SEGURANÇA EM CAMADAS

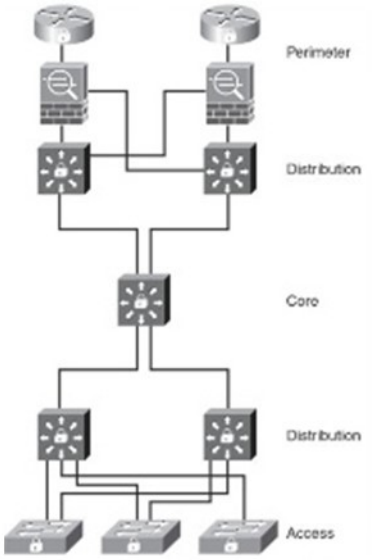
A segurança em camadas é a mais recomendada e a abordagem mais escalável para a segurança de redes. Apenas um único mecanismo não pode ser confiado para garantir a segurança de um sistema. Para proteger a infraestrutura de uma organização, é necessário aplicar a segurança em diversas camadas. Esta abordagem é também chamada como "defesa em profundidade". A ideia é criar múltiplos sistemas para que uma falha em um deles não deixe a rede vulnerável. Adicionalmente, num sistema de camadas, as vulnerabilidades podem ser limitadas e contidas na camada afetada [BHAJI Y. – 2008].

As soluções de segurança atuais se movimentam em direção a sistemas de várias camadas de rede e não apenas aos dispositivos de borda. Atualmente, a recomendação é a inclusão de Sistemas de Prevenção de Intrusão (Intrustion Prevention System – IPS) dentro e fora de redes privadas. Firewalls são colocados entre vários segmentos de negócio ou departamentos dentro de uma mesma organização, dividindo a rede em grupos lógicos e

aplicando perímetros de defesa em cada segmento ou departamento. Neste modelo multi-perímetro, cada segmento pode possuir diferentes camadas de defesa dentro dele.

Um perímetro de segurança efetivo se tornou muito importante com o passar dos anos. Não é possível confiá-lo apenas aos mecanismos de defesa tradicionais (Firewalls e Sistemas de detecção de intrusos – IDS). Aplicações web, acessos sem fio, redes interconectadas e VPN's fizeram do perímetro um conceito muito complexo.

Uma abordagem de camadas requer a implantação de diferentes soluções de segurança em diferentes espectros da rede. Um conceito similar são as "ilhas de segurança". Para implantá-las, não se pode restringir aos conceitos de perímetro de segurança. Não depende de um ou de outro método para garantir a eficácia da segurança. Deve-se pensar em perímetros, distribuição, núcleo e acessos.



Fonte: Network Security Technologies and Solutions

Figura 03: Camadas de Segurança

Esta abordagem é relacionada com as tecnologias de um ambiente e a complexidade de cada uma das tecnologias em cada camada. A complexidade vem de diferentes protocolos, aplicações hardware, e mecanismos de segurança que funcionam em um ou mais das sete camadas do modelo OSI (Open System Interconnection) – Aplicação, Apresentação, Sessão, Transporte, Rede, Enlance e Física. Num determinado ambiente há diferentes níveis e diferentes tipos de ataques podem ocorrer, sendo necessário identificar as contramedidas necessárias.

O modelo de referência OSI foi criado para habilitar que diferentes camadas trabalhem independentemente das outras. A abordagem foi desenvolvida para acomodar mudanças nas tecnologias envolvidas. Cada camada OSI é responsável por uma função específica dentro do conjunto de rede, com fluxos de informação se movimentando entre as camadas enquanto os dados são processados. Infelizmente, isso significa que, se uma camada é invadida, as comunicações são comprometidas sem que as outras camadas sejam alertadas. A segurança é tão forte quanto seu elo mais fraco. Qualquer uma das camadas OSI podem ser este elo [STALLINGS W. – 2011].

Figura 04: Modelo de Camadas OSI [The OSI Model: Understanding Seven Layers of Computer Networks – 2015]

Desta forma, podemos entender que o Apache Spot funcionaria entre as camadas 3 e 7, pois não engloba segurança em termos binários (entendimento bit-a-bit), nem de acesso aos meios, mas sim, a partir da camada de rede com protocolo IP, conexões ponto a ponto, comunicação entre servidores, representação de dados e processos de rede para aplicações.

1.6 RODA DE SEGURANÇA

A segurança de redes é um processo contínuo construído ao redor de uma política de segurança corporativa. A Roda de Segurança é um processo recursivo em curso com o objetivo de perfeição: Construir uma infraestrutura de rede segura. O paradigma incorpora os seguintes passos [BHAJI Y. – 2008]:

1. **Desenvolver uma política de segurança:** Uma forte política de segurança deve ser claramente definida, implantada e documentada, além de ser simples o suficiente para que os usuários possam conduzir os negócios dentro dos parâmetros.
2. **Fazer a rede segura:** Implantar soluções de segurança (autenticação, criptografia, firewalls, prevenção de intrusões e outras técnicas) para parar ou prevenir acessos e atividades não autorizadas e para proteger os sistemas de informação.
3. **Monitorar e reagir:** Esta fase detecta violações na política de segurança. Envolve auditoria de sistemas e detecção de intrusões em tempo real e soluções de prevenção. Também valida as implantações de segurança do passo anterior.
4. **Testar:** Este passo valida a efetividade da política de segurança através da auditoria dos sistemas e escaneamento de vulnerabilidades e testes nas soluções de segurança.
5. **Gerenciar e Melhorar:** Usar as informações de monitoramento e fases de teste para melhorar a implantação de segurança. Ajustar as políticas de segurança ao passo que vulnerabilidades e riscos são identificados.

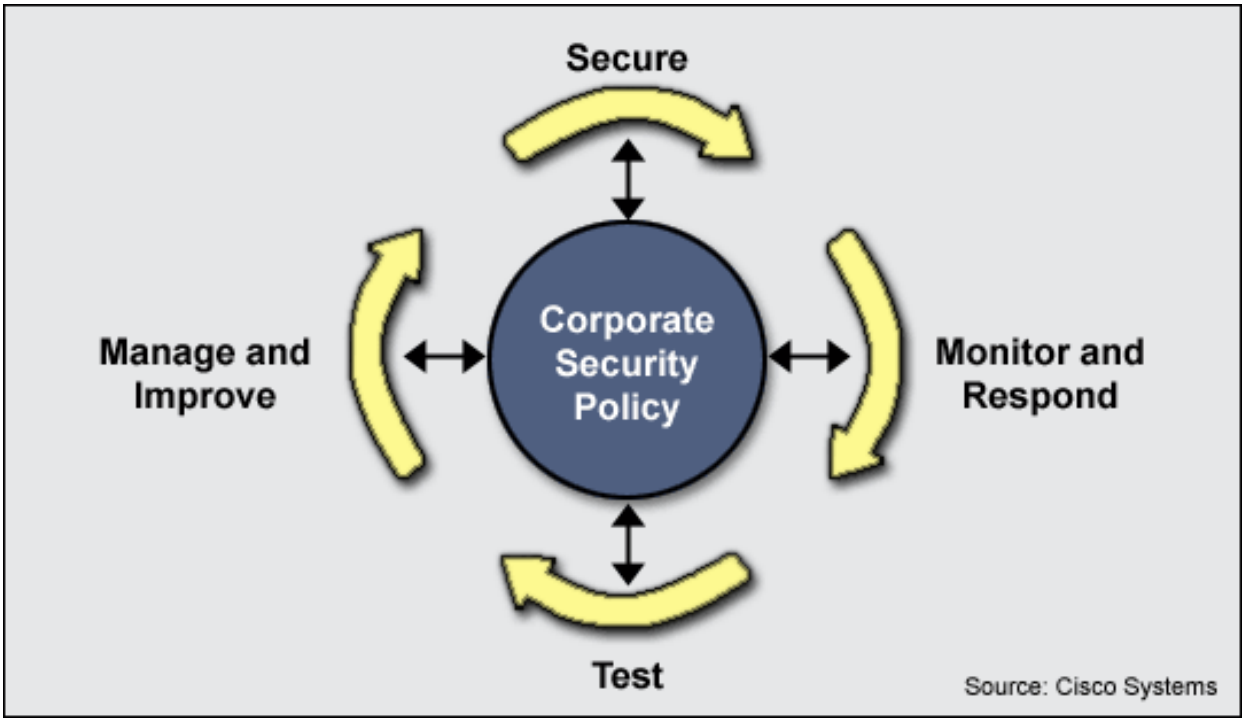


Figura 05: Roda de Segurança CISCO [CISCO - 2011]

1.7 CLASSES E VETORES DE ATAQUE

Uma das possíveis divisões de ataques à sistemas [STALLINGS W. – 2011] seria classificar por classes (diferentes categorias) e vetores (técnicas específicas).

1.7.1 CLASSES DE ATAQUES

Os principais tipos de ataque são:

- **Reconhecimento:** São os primeiros passos de um processo de intrusão e envolve descobertas não autorizadas e mapeamento de sistemas, serviços ou vulnerabilidades. Estas técnicas de mapeamento e descoberta são conhecidas como escaneamento e enumeração. Ferramentas, comandos e utilitários comuns são utilizados para este processo, como “ping”, “telnet”, “nslookup”, “finger”, “rpcinfo”, “Explorador de Arquivos”, “srvinfo”, “dumpacl”, “Sniffer”, “SATAN”, “SAINT”, “NMAP” e “netcat”.
- **Acesso:** Ataques de acesso se referem à manipulação de dados não autorizada que fornece ao atacante acesso aos sistemas ou privilégios na máquina em questão. Recuperação de dados não autorizada é simplesmente o ato de ler, escrever, copiar ou mover arquivos que não são permitidos para o intruso. Algumas atividades comuns são realizadas nesta fase, incluindo explorar senhas, acessar informações confidenciais, exploração de serviços mal configurados ou não gerenciados, acessar registros remotos, abuso de uma relação confiável, roteamento de IP e compartilhamento de arquivos.
- **Negação de Serviço (Denial of Service – DoS)**: Um ataque DoS acontece quando um atacante intencionalmente bloqueia, degrada, desabilita ou corrompe redes, sistemas ou serviços, com a intenção de negar o serviço aos usuários autorizados. O ataque é criado para impedir a disponibilidade de um recurso para os usuários autorizados com a desativação do sistema ou diminuindo a velocidade do mesmo até o ponto em que se torna inutilizável. Ataques comuns DoS incluem TCP SYN floods, ICMP ping floods e sobrecarga de buffer.

1.7.2 VETORES DE ATAQUE

Vetores de ataque são rotas ou métodos utilizados para entrar em um computador ou sistemas de rede para encontrar aberturas inesperadas para uso indevido. São geralmente classificados [STALLINGS W. – 2011]:

- **Vírus:** É um software malicioso ou um pedaço de código que causa um evento negativo não antecipado e normalmente é capaz de causar danos aos dados ou outros programas no sistema infectado.
- **Worms:** É um software malicioso que se replicada, similarmente ao vírus. Podem residir na memória ativa de um sistema e são capazes de se duplicar e se propagar de um computador para uma rede inteira. Worms geralmente são designados para explorar as capacidades de transmissão de arquivos, como o e-mail por exemplo.
- **Cavalo de Tróia (Trojans):** Programa malicioso que pretende ser uma aplicação benigna. São comumente assimilados como programas inofensivos que escondem atividades maliciosas, como captura de teclado (para capturar senhas dos usuários que digitam na máquina ou qualquer outra informação sensível sem o conhecimento do usuário).
- **Sobrecarga de Buffer:** Buffer são localizações de memória num sistema que são utilizados para armazenar dados. Uma sobrecarga no buffer ocorre quando um programa tenta armazenar dados no buffer, maior do que a capacidade alocada, fazendo com que os dados sejam perdidos por falta de armazenamento.
- **IP Spoofing:** Ocorre quando um intruso tenta apresentar um IP de um endereço confiável para ganhar acesso a recursos específicos numa rede confiável. É um dos atos mais comuns de camuflagem online.
- **Address Resolution Protocol (ARP) Spoofing:** Ocorre quando um intruso tenta esconder o endereço de Hardware (endereço MAC) para personificar um servidor confiável. Este é um dos passos iniciais que ajuda a maioria dos outros ataques.
- **Man-in-the-middle (TCP hijacking):** Neste ataque, o intruso intercepta uma comunicação legítima entre dois pontos e pode modificar ou controlar a sessão TCP sem o conhecimento do remetente e do receptor. O TCP hijacking usa como alvo as aplicações baseadas em TCP, como Telnet, FTP, SMTP (e-mail) ou HTTP. Um intruso pode, além do que já foi descrito, utilizar um programa para observar a conversação.

- **Ping Sweeps:** Também conhecido como Internet Control Message Protocol (ICMP) sweep, é uma técnica de escaneamento utilizada para determinar servidores ativos numa rede. É uma varredura que consiste em requisições ICMP ECHO enviada a múltiplos computadores. Caso a máquina responda, o atacante já consegue legitimar que é um servidor ativo. Este processo é muito utilizado na fase de reconhecimento para um futuro ataque.
- **Escaneamento de portas:** É um método utilizado para enumerar quais serviços estão ativos num sistema. Um intruso envia requisições aleatórias em diferentes portas e, quais responderem à essas requisições, será possível confirmar as portas ativas no modo de escuta. O atacante pode planejar explorações em quaisquer vulnerabilidades conhecidas nessas portas. Um "scanner" de portas é um software designado para buscar portas abertas numa rede. Também é uma técnica primária para reconhecimento de ambiente em busca de vulnerabilidades.
- **Sniffing:** Um sniffer é um software que utiliza um adaptador de rede de modo promíscuo para capturar todos os pacotes de rede que são transmitidos na rede.
- **Flooding:** Ocorre quando uma quantidade excessiva de dados desnecessários é enviada, resultando numa disrupção da disponibilidade de dados.
- **Ataques DoS/DDoS:** Na maioria dos casos, um ataque DoS serve para privar os usuários legítimos à serviços ou recursos. Normalmente não resultam em invasões ou roubo de informações ilegal, mas são arquitetados para prevenir o acesso de usuários autorizados, sobrecarregando a vítima com um volume excessivo de pacotes. Distributed DoS (Negação de Serviço Distribuída) amplifica os ataques DoS com um número grande de sistemas compromissados e coordenados para sobrecarregar os alvos.

1.7.3 AVALIAÇÃO DE RISCO

É imperativo auditar a rede e avaliar a postura de segurança para os riscos e ameaças no ambiente, com o objetivo de determinar as probabilidades e ramificações de uma falha de segurança. Este deveria ser um processo iterativo que é possível avaliar e ranquear cada ameaça e identificar as técnicas de mitigação de acordo. Os ataques às redes mais comuns [CISCO – 2011]:

- 75%-80% não são detectados;
- 15%-20% são instigados pelo mundo externo;
- 80%-85% são realizados internamente – pessoas com autorização;
- 80%-90% são ataques vingativos por scripts de amadores;
- 10% são um tipo mais sério de ataque DDoS;
- 1%-5% afetam a infraestrutura diretamente.

O modelo de ameaças envolve identificar e ranquear as ameaças de acordo com a probabilidade e o dano que elas podem causar potencialmente.

- **1° Passo:** Identificar vulnerabilidades, ameaças, potenciais vetores de ataque e o potencial impacto na rede e performance.
- **2° Passo:** Categorizar cada ameaça por criticidade, ou seja, quanto dano um ataque desta natureza pode causar e qual é a probabilidade de ocorrência.
- **3° Passo:** Criar uma métrica para a criticidade.
- **4° Passo:** Identificar a técnica ou tecnologia apropriada para mitigar cada ameaça. Cada ameaça possui uma técnica de mitigação específica com várias opções.
- **5° Passo:** Repetir todo este processo continuamente para garantir a segurança de sua infraestrutura aos novos riscos e ataques não identificados.

Não existem soluções perfeitas para todos os problemas de segurança. A cada informação que flui dentro de uma rede, existem diversas possibilidades de caminhos que ela pode percorrer e novos desafios surgem constantemente para a área de segurança [BHAJI Y. – 2008].

1.8 OBJETIVOS

O objetivo deste trabalho será explorar a ferramenta Apache Spot (antigo Open Network Insight) numa rede testes e apresentar suas funcionalidades e possíveis benefícios para a área de segurança da informação, pelo fato de ser apresentado como uma nova opção no mercado para monitoramento de possíveis ameaças na rede e escalabilidade para ambientes complexos. Seu código utiliza recursos do Hadoop e é aberto para desenvolvimento e contribuição da comunidade, com grande potencial para o mundo corporativo e acadêmico na área de segurança da informação.

Uma das contribuições que foi considerada neste trabalho será também apresentar todos os passos realizados para a instalação do ambiente em uma infraestrutura em nuvem, visto que, as ferramentas de código-aberto muitas vezes não são amigáveis aos usuários, sendo complexa sua execução.

Considerando toda a introdução teórica de segurança abordada anteriormente, as próximas seções irão abordar os fundamentos que envolvem o projeto desenvolvido. Esta é uma ferramenta capaz de processar volumes enormes de dados (por isso considerado como Big Data) que envolvem informações referentes à trafego de rede.

1.9 BIG DATA

Serviços como redes sociais, análises da web, e-commerce inteligente, usualmente precisam gerenciar dados numa escala muito maior do que os bancos de dados tradicionais. Com o aumento das escalas e da demanda, também aumenta a complexidade. Felizmente, escalabilidade e simplicidade não são mutualmente exclusivos. Em vez de se usar uma tecnologia em destaque, uma abordagem diferente é necessária. Sistemas de Big Data utilizam muitas máquinas trabalhando em paralelo para armazenar e processar dados, o que introduz desafios fundamentais para a maioria dos desenvolvedores. Big Data mostra como construir esses sistemas utilizando uma arquitetura que garante as vantagens de um hardware clusterizado com novas ferramentas desenhadas especificamente para capturar e analisar dados em grande escala.

Durante este trabalho, um dos princípios mais relevantes é a utilização de ferramentas de Big Data, devido à necessidade de uma ferramenta altamente escalável em termos de volumes de dados que serão analisados e que podem ser provenientes de diferentes tipos de fontes.

Uma das características desta tecnologia é ser altamente versátil para a recepção de dados, porém de uma maior complexidade. Projetos de Big Data costumam seguir de 5 a 6 fases:

1. Instalação de infraestrutura para recepção de dados (ex: Hadoop);
2. Desenvolvimento de processos de ingestão de dados crus (ex: Python, Spark, Scala, Shell Script, Sqoop);
3. Transformações e enriquecimento de informações (ex: Python, Spark, Scala);
4. Análise dos dados ingeridos e transformados (utilização de linguagens SQL Like);

5. Criação de uma camada de visualização (ex: Qlikview, Tableau, PowerBI);
6. Modelagem estatística e preditiva (ex: LDA, kNN, Churn, etc).

A fase de modelagem estatística não necessariamente é adotada e muitas vezes também pode não trazer os resultados esperados, pois eventualmente as variáveis utilizadas não são suficientes ou o modelo não é adequado para a situação proposta.

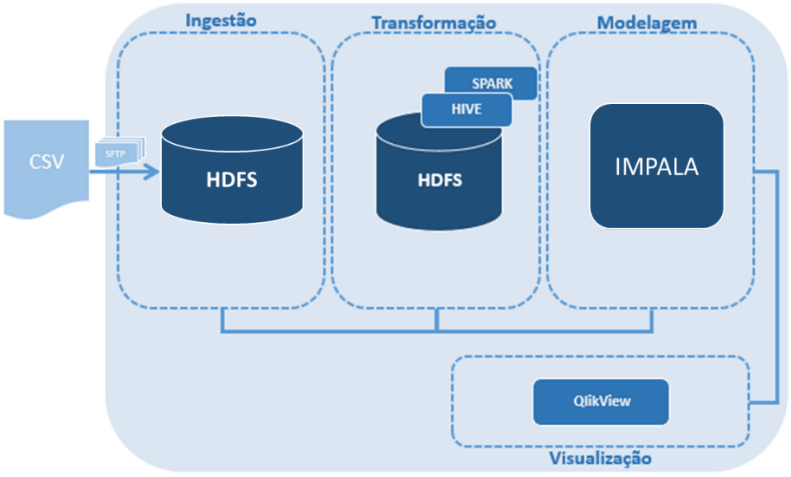


Figura 06: Exemplo de projeto de Big Data

Na figura 06 ilustrada acima, temos um exemplo simples das fases de desenvolvimento de um projeto de Big Data. Arquivos CSV são ingeridos para o Hadoop Distributed File System (HDFS), onde são transformados com a utilização do Spark e modelados com o Impala e visualizados através do Qlikview. Neste exemplo, não necessariamente seria utilizado um modelo matemático, mas apenas uma separação dos dados na criação de determinados indicadores.

O exemplo acima é apenas ilustrativo para o entendimento das tecnologias de Big Data. Os componentes considerados fazem parte do ecossistema Hadoop (Hive, Spark, HDFS) e o Impala faz parte da distribuição Cloudera do Hadoop e o QlikView é um visualizador de dados. Não serão detalhados todos os componentes, pois não é o foco deste trabalho.

No contexto de crescimento de tecnologias de nuvem (Amazon Web Services, Azure, Google Cloud, etc), talvez uma ferramenta como o Apache Spot seja de grande auxílio para avaliação do tráfego de rede. Muitas instâncias são colocadas em funcionamento e desligadas frequentemente, gerando imensos tráfegos nas redes de nuvem e podem ter os mais diversos tipos de ameaças nestes ambientes.

1.10 CLOUDERA® DISTRIBUTION OF HADOOP

O Apache Hadoop é um tipo de plataforma de dados capaz de processar distributivamente grandes conjuntos de dados através de clusters de computadores utilizando simples modelos de programação. Ele foi desenvolvido para escalar de servidores isolados até milhares de máquinas, cada uma oferecendo armazenamento e computação local. Mais do que confiar no hardware para garantir a alta disponibilidade, o framework é desenhado para detectar e lidar com falhas na camada da aplicação.

A distribuição Cloudera® do Apache Hadoop constitui todo o seu ecossistema preparado para o armazenamento e processamento de uma larga escala de dados baseado no código Open Source da Apache. Devido às necessidades das empresas que utilizam a plataforma, a Cloudera® uniu todos os componentes necessários para o manuseio de grandes volumes de dados e presta suporte para quaisquer problemas apresentados em seu código-fonte, para que seus dados estejam protegidos e seu processamento de dados garantido, por não ser possível depender da comunidade para resolução de problemas inesperados em um tempo de resposta adequado para o negócio [Documentação oficial Cloudera Enterprise Data Hub – 2018].

O objetivo deste trabalho não é explorar a plataforma Hadoop, mas é necessário enfatizar a dependência deste ecossistema para o Apache Spot. Abaixo na figura 07, está uma arquitetura padrão de um ambiente Cloudera:

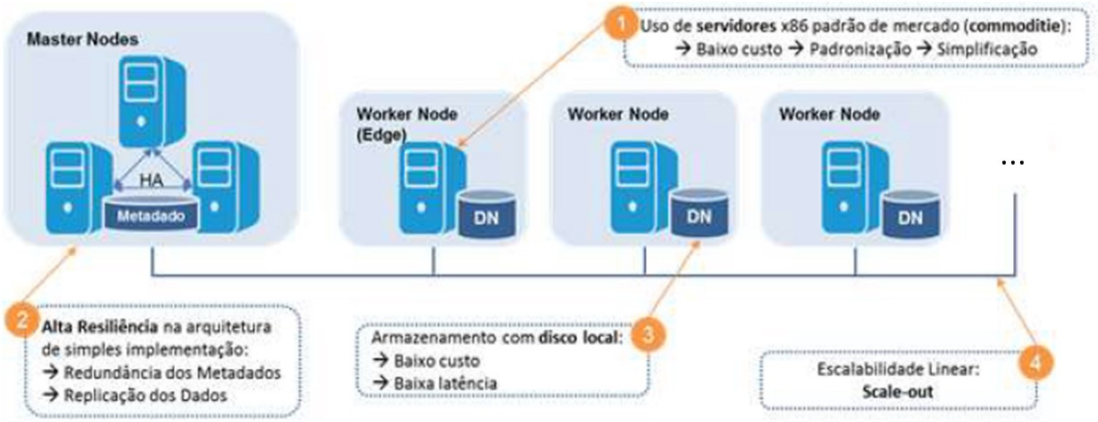


Figura 07: Arquitetura de infraestrutura Cloudera [Documentação oficial Cloudera Enterprise Data Hub – 2018]

1.11 APACHE SPOT

Com redes e datacenters dinâmicos, o Apache Spot foi desenvolvido como uma ferramenta avançada para detecção de ameaças, através de modelos analíticos baseados em Big Data, que podem ser escalados em soluções de nuvem, para prover visões e ações contra pacotes maliciosos trafegando na rede. Com esta ferramenta, é possível analisar bilhões de eventos para detectar ameaças não conhecidas, atacantes internos e ganhar um novo nível de visibilidade da rede.

Não existem informações sólidas a respeito dos criadores desta ferramenta e a primeira publicação encontrada foi no Twitter no dia 29/02/2016). Inicialmente batizada com o nome Open Network Insight, foi desenvolvida por uma pequena equipe da Intel® dedicada com habilidades de segurança da informação e tecnologias de Big Data para criar uma solução única de segurança, baseado em Hadoop e recentemente se tornou um projeto Apache para contribuição da comunidade da Internet com o nome Spot.

A solução combina o processamento das tecnologias de Big Data, aprendizado de máquina escalável e um modelo analítico único em segurança para colocar potenciais ameaças em evidência. Abaixo na figura 08, temos uma sumarização de problemas de segurança:

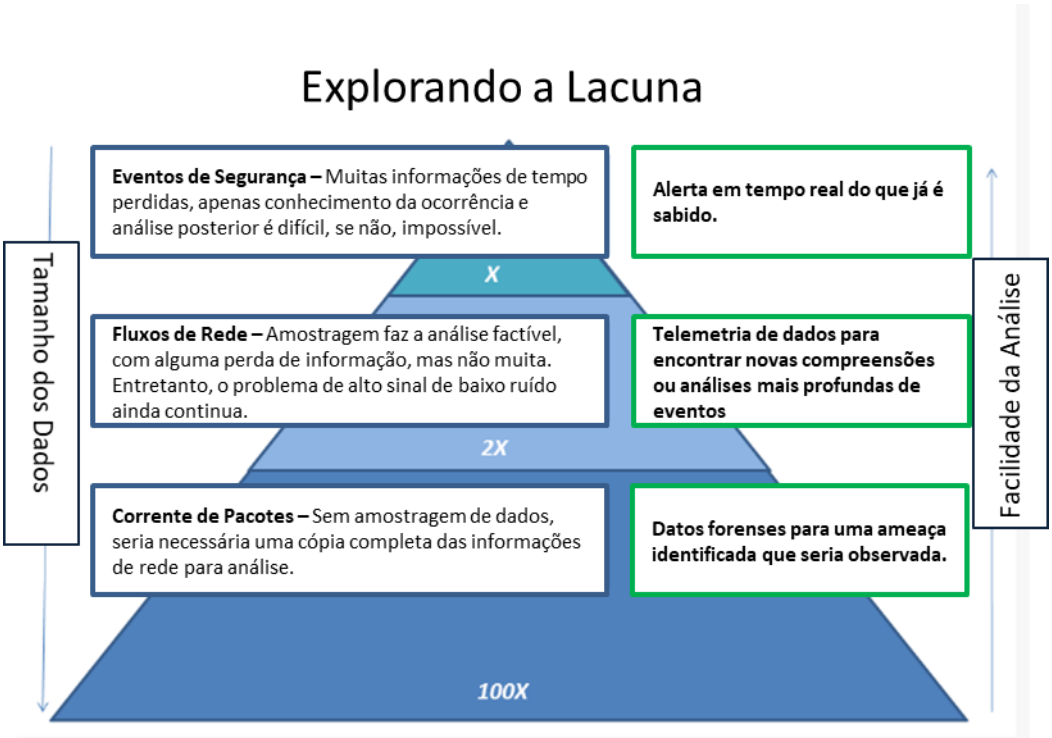


Figura 08: Sumarização dos problemas de Segurança por Grant Babb [Documentação Oficial Apache Spot - 2018]

O Apache Spot é uma ferramenta com código aberto, com soluções analíticas para fluxos e pacotes na rede. O objetivo é aprender com os eventos para prever ameaças e tomar decisões automatizadas em momentos críticos e que pessoas em seu lugar teriam um tempo de reação muito maior, assim como saber os motivos dos ataques e como devem ser tratados posteriormente, fato que não ocorre atualmente.

Apesar de ter sido disponibilizada recentemente, o Apache Spot traz diversas características muito interessantes para o mundo da segurança da informação e aberto para todos os usuários, diferentemente das outras soluções de segurança oferecidas no mercado atualmente. São elas:

- Perímetro de fluxos de dados;
- Monitoração de pacotes DNS;
- Monitoração de Proxy;
- Fluxos internos.

O Spot utiliza o aprendizado de máquina como um filtro para separar o tráfego de informações maliciosas das benignas e caracterizar unicamente o tráfego na rede. São características de análise da ferramenta:

- **Pacotes DNS Suspeitos:** Capaz de executar uma inspeção profunda no tráfico de DNS para construir um perfil de cargas úteis de DNS. Após visualizar, normalizar e conduzir as buscas de padrões, o analista possuirá uma lista das ameaças mais prováveis no tráfico de DNS.
- **Incidentes e Respostas:** Dado um endereço IP, o Spot consegue juntar todas as características sobre a comunicação associada (como uma rede social do IP) e cria uma linha do tempo das conversas originárias daquele IP.
- **Conexões suspeitas:** Utiliza aprendizado de máquina para construir um modelo de máquinas numa rede e seus padrões de comunicação. As conexões entre as máquinas com menor probabilidade são visualizadas e filtradas pelo ruído, para então serem reconhecidas como padrões. Os padrões resultantes são provavelmente centenas de pacotes dentro de bilhões de outros.

- **Storyboard:** Após uma investigação de ameaça, ainda é necessário comunicar o evento que ocorreu para a organização com o dashboard disponibilizado na ferramenta.

1.11.1 FUNCIONAMENTO DO APACHE SPOT

Com a figura 09 abaixo apresentada, é possível ter um breve entendimento de forma **simples e concisa sobre o funcionamento da ferramenta:**

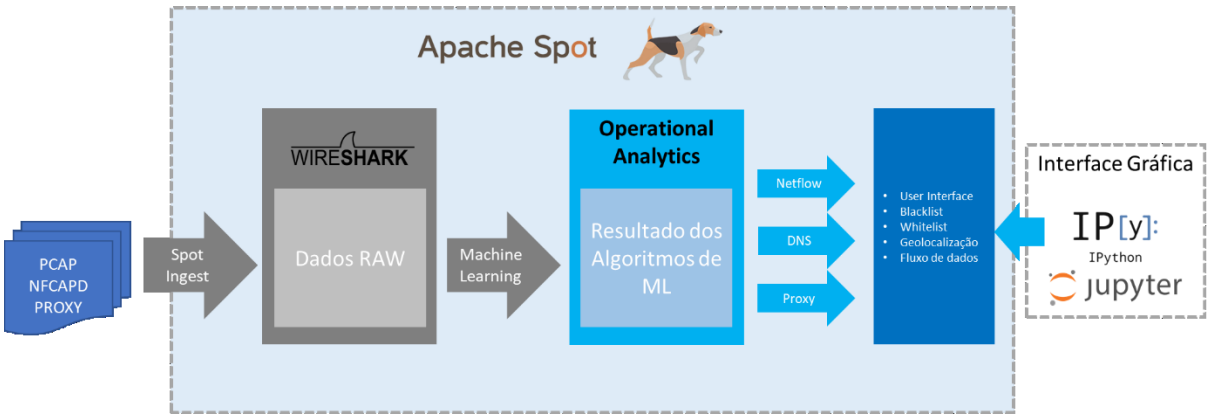


Figura 09: Diagrama funcional Apache Spot

1.11.2 SPOT INGEST

O processo de funcionamento do Apache Spot se inicia através de um processo de ingestão através de um framework disponibilizado pela ferramenta. Os dados de tráfego de rede, DNS ou Proxy são capturados e/ou transferidos para o cluster Hadoop através deste componente, que utiliza o Wireshark para leitura das informações recebidas (não entraremos no mérito do funcionamento do Wireshark, mas basta saber que é uma ferramenta mundialmente conhecida para análise de protocolos de rede em todas as camadas OSI).

Uma característica interessante apresentada por este componente é a utilização de um Open Data Model, capaz de normalizar diversos tipos de dados que são ingeridos, visto que cada ferramenta de captura traz informações de forma diferente. Abaixo na figura 10 está ilustrado como é o funcionamento da camada de ingestão da ferramenta:

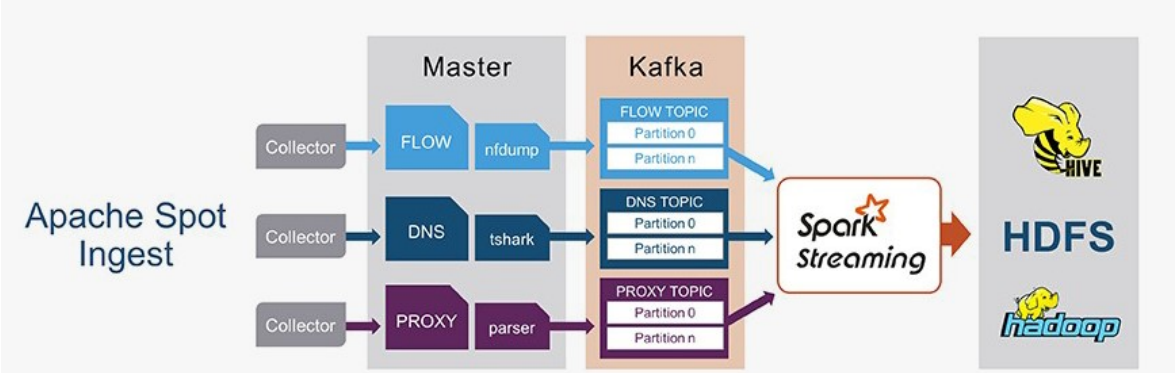


Figura 10: Diagrama funcional da camada de ingestão [Site Oficial Apache Spot]

1.11.3 MACHINE LEARNING

O segundo passo do processo que será seguido é o processamento das informações (dados RAW – ou crus, traduzido para o Português) utilizando uma aplicação Spark que é baseada no algoritmo de LDA (Latent Dirichlet Allocation).

O Spark é um framework construído para criação de aplicações que serão utilizadas em ambientes distribuídos com altos volumes de dados, com a capacidade de ser veloz, versátil e escalável.

Segundo o trabalho da Universidade de Stanford (M. Blei, David / Y. Ng, Andrew / I. Jordan, Michael - 2003), este algoritmo é um modelo probabilístico generativo para coleções de dados discretos, como texto, por exemplo. É um modelo Baysesiano hierárquico de três níveis, em que, cada item de uma coleção é modelado como uma mistura finita sobre um conjunto de tópicos probabilísticos subjacentes.

Fundamentalmente, modelos de tópicos descobre relações entre documentos e termos, gerando padrões latentes que sejam significativos para o entendimento dessas relações. Um exemplo seria classificar um conjunto de termos como relevantes para determinados temas. O LDA é um modelo probabilístico no qual é descrito como os documentos são gerados. Neste modelo, as variáveis observáveis são os termos de cada documento e as variáveis não-observáveis são as distribuições de tópicos.

No processo generativo, o resultado da amostragem de Dirichlet é usado para alocar as palavras de diferentes tópicos e que preencherão os documentos.

O Apache Spot aplica o algoritmo no tráfego de rede convertendo as entradas em palavras através de agregações e discretizações. Desta maneira, documentos correspondem à endereços IP, palavras à entradas (relacionadas à um determinado IP) e tópicos a perfis de atividades de rede comuns.

Desta forma, infere um modelo probabilístico para o comportamento de rede de cada endereço IP. Cada entrada de rede é assinalada à uma probabilidade estimada (pontuação) pelo modelo e os eventos com menor pontuação serão marcados como suspeitos para uma posterior análise. [Documentação Apache Spot]

Este, provavelmente é o componente mais importante presente no Apache Spot, visto que é através desta inteligência que os dados são analisados e identificados como possíveis ameaças ao ambiente monitorado. Uma das funcionalidades identificadas no componente de Aprendizado de Máquina é que, para o aprendizado supervisionado (o qual requer uma interação do usuário para melhorar a análise), ele replica os dados de ameaça identificados milhares de vezes para que este dado seja considerado uma vulnerabilidade no ambiente.

1.11.4 OPERATIONAL ANALYTICS E INTERFACE DO USUÁRIO

Este componente é o responsável por apresentar todos os resultados provenientes do algoritmo de Machine Learning descrito anteriormente e coloca-los num contexto enriquecido, com filtragem de ruído e heurística, produzindo uma lista mais adequada para mostrar padrões e possíveis ameaças.

Desta forma, há uma redução enorme nos pacotes que serão analisados, de forma a amplificar as chances de se encontrar uma ameaça em meio a tanta informação [Apache Spot – 2018].

Através desses resultados, o Operational Analytics alimenta as seguintes funcionalidades da ferramenta:

- Interface do usuário;
- Blacklist e Whitelist;
- Geolocalização;
- Fluxo de dados interno e externo.

A interface gráfica é construída com o Ipython e Jupyter Notebook, conforme ilustrado na figura 11:

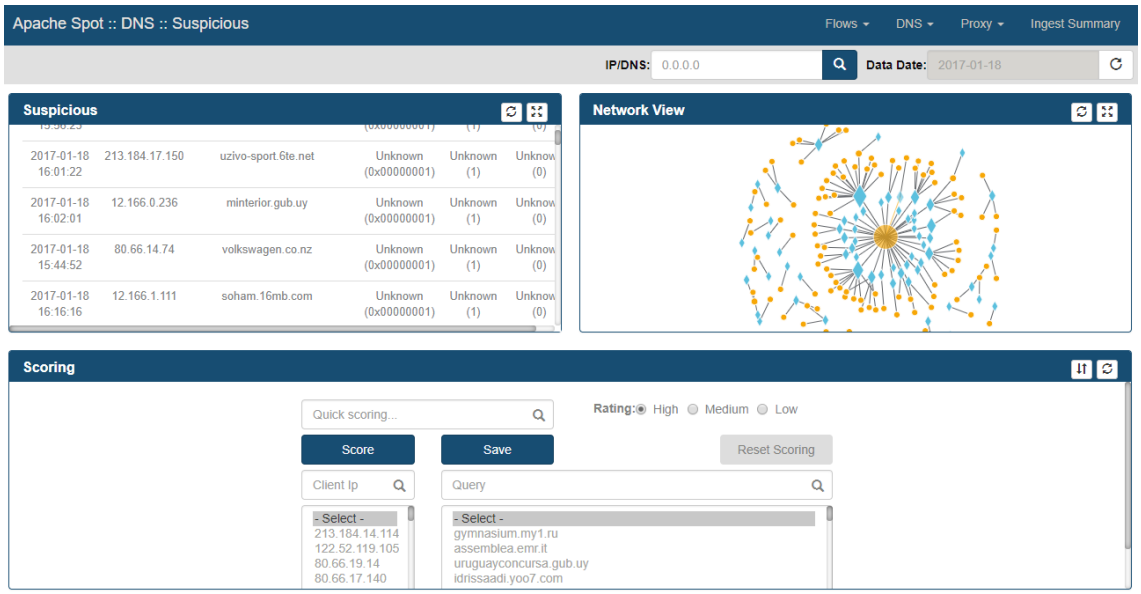


Figura 11: Interface produzida pelo Apache Spot

2 METODOLOGIA

Para a construção do projeto, a seguinte imagem de referência (figura 09) já apresentada anteriormente será utilizada, de forma a guiar todas as necessidades da ferramenta:

Para o projeto, serão analisadas todas as características presentes no Apache Spot num ambiente controlado e os seguintes passos serão seguidos:

- Verificação de requisitos de hardware e software;
- Instalação de Sistema Operacional;
- Ajustes de Sistema Operacional;
- Instalação da ferramenta Apache Spot;
- Instalação dos componentes da ferramenta;
    - Operational Analytics;
    - Machine Learning;
    - Ingest;
- Interface Gráfica;
- Análise do fluxo de dados da ferramenta;
- Busca de pacotes disponibilizados na Internet para testes;
- Testes e análises.

2.1 PRÉ-REQUISITOS PARA INSTALAÇÃO DO APACHE SPOT

Para realizar a instalação do Apache Spot, alguns componentes do ecossistema Hadoop são necessários e serão utilizados pela ferramenta para seu total funcionamento:

- Sistema Operacional CentOS 7 (Não mandatório);
- Cloudera Distribution of Hadoop 5.4;
- HBase (Banco de dados não-relacional);
- Hive (Data Warehouse);
- Hue (Visualizador de Dados);
- Impala (Banco de Dados Analítico para Structured Query Language);
- Kafka (Plataforma de streaming de dados);
- Oozzie (Plataforma para agendamento de fluxo de trabalho);
- Zookeeper (Serviço de manutenção de configurações para sistemas distribuídos);

- Spark (Motor para processamento em larga escala);
- YARN (Gerenciador de recursos do Hadoop);
- HDFS (Sistema de arquivos tolerante a falhas).

Para a instalação, a arquitetura de referência apresentada pela Apache na figura 12:

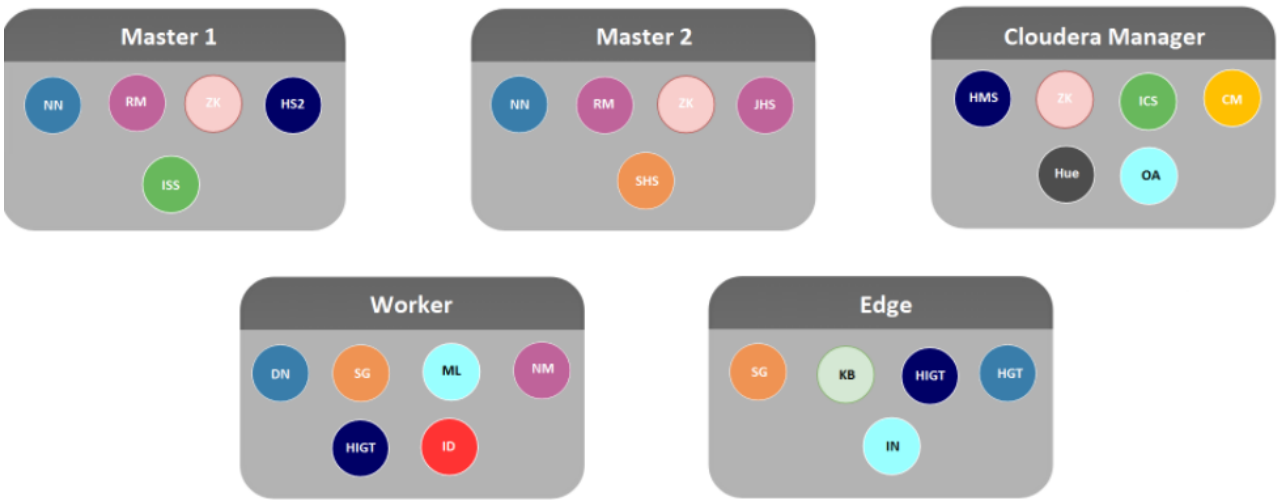


Figura 12: Arquitetura de Software de referência para o Apache Spot

Especificações:

- NN – Name Node;
- DN – Data Node;
- RM – Resource Manager;
- JHS – Job History Server;
- SHS – Spark History Server;
- NM – Node Manager;
- ICS – Impala Catalog Server;
- ISS – Impala State Store;
- HMS – Hive Metastore Server;
- HS2 – Hive Server 2;
- HGT – HDFS Gateway;
- HIGT – Hive Gateway;
- CM – Cloudera Manager;
- ZK – Zookeeper;
- SG – Spark Gateway;

- OA – Operational Analytics;
- ML – Machine Learning;
- IN – Ingest;
- ID – Impala Daemon;
- KB – Kafka Broker;

Entretanto, por se tratar de um ambiente de testes, não há a necessidade de servidores tão robustos (128-512GB RAM) pela demanda de tempo para as análises e ingestões de dados. Para este trabalho, foi apenas utilizada três máquinas na nuvem no modelo "*m4.xlarge*" da Amazon Web Services, com as seguintes especificações:

- 4 vCPUs (Processador Intel Xeon E5-2686 v4 Broadwell 2,3 GHz);
- 16 GB RAM;
- Largura de banda dedicada de 750 Mbps;
- 1 disco rígido de 100 GB (Sistema Operacional);
- 1 disco rígido de 300 GB (Aplicação);

Todos os componentes da distribuição Cloudera® do Hadoop foram instaladas neste mesmo servidor.

2.2 INSTALAÇÃO DO SISTEMA OPERACIONAL

Como o servidor utilizado foi criado na AWS, não foi necessária uma instalação específica para o Sistema Operacional. Apenas selecionado no console da AWS qual era o SO desejado e automaticamente a instância foi criada com a instalação correta, conforme a figura 13 a seguir.

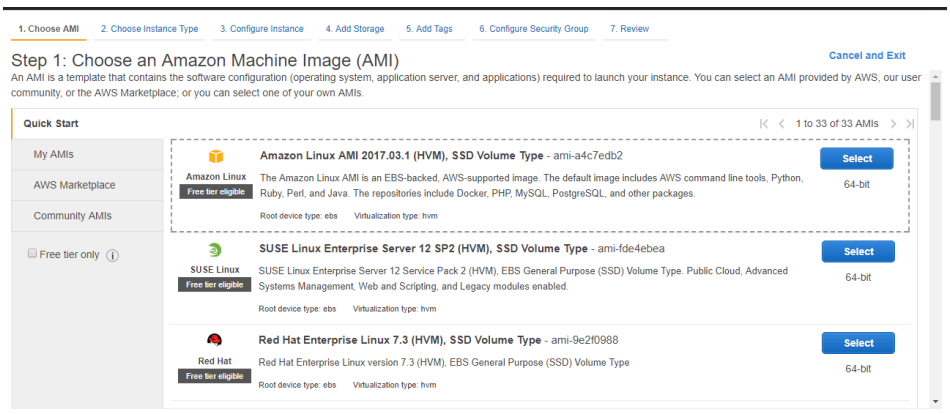


Figura 13: Configuração de instância AWS

Configurações da instância:

- Número de instâncias: 03 instâncias
- Rede: Automática
- Subrede: Automática
- Auto-atribuição de IP Público: Sim
- Grupo de Colocação: -
- Regra IAM: -
- Comportamento de desligamento: Não
- Proteção de finalização: Não
- Monitoramento: Não

2.3 AJUSTES DO SISTEMA OPERACIONAL

Para que fosse realizada a instalação, alguns ajustes no sistema operacional foram realizados durante a instalação e estão descritos no item a seguir.

2.4 INSTALAÇÃO DA DISTRIBUIÇÃO CLOUDERA APACHE HADOOP

Para que fosse realizada a instalação do Apache Spot, era pré-requisito a instalação de um ambiente Hadoop. A distribuição Cloudera foi escolhida devido à recomendação nos documentos oficiais da ferramenta.

O passo-a-passo foi listado e explicitado, entretanto os lançamentos de novas versões do Cloudera ocorrem aproximadamente a cada 2 meses, gerando sempre pequenas alterações que devem ser visualizadas no site oficial.

Os seguintes passos foram utilizados para a instalação dos componentes necessários para o Apache Spot:

1. Desabilitar SELINUX (Firewall de aplicação que pode bloquear as aplicações e elas não funcionarão da maneira adequada);
2. Desabilitar Swap (Caso o Swap esteja habilitado, a performance do ambiente pode ser prejudicada pelo desempenho apresentado pelos discos);
3. Configurar número de arquivos (O ambiente do Hadoop precisa ler e abrir um número de arquivos muito maior do que vem configurado por padrão no SO;

4. Configurar hostname (Resolução de nome precisa estar correta para os nós que compõe o cluster);
5. Configurar /etc/hosts (Resolução de nome precisa estar correta para os nós que compõe o cluster);
6. Desabilitar Firewall local (Objetivo é não afetar acessos entre as máquinas que compõe o Cluster);
7. Apresentar discos da AWS (Seguindo o padrão determinado pela Cloudera para melhor performance);
8. Montar disco do HDFS (Este será o disco em que os dados serão armazenados no ambiente);
9. Editar /etc/fstab e adicionar novo ponto de montagem para o HDFS (Na hora de iniciar o cluster, os discos estarão apresentados);
10. Atualizar e instalar pacotes (Instalação da aplicação no ambiente);
11. Instalar banco de dados MySQL (Banco de dados que será utilizado para guardar os metadados, dados do Cloudera Manager, informações de segurança, etc);
12. Editar my.cnf (Necessário colocar as configurações do MySQL nas melhores práticas para evitar falhas e perda de desempenho);
13. Desabilitar links simbólicos (Para prevenir riscos de segurança);
14. Colocar o diretório "/log_bin" num disco com espaço livre (Garantir que o diretório de replicação do MySQL esteja separado do disco do SO);
15. Alterar o proprietário do diretório para o usuário "mysql" (Para apenas o usuário do MySQL ter acesso ao banco);
16. Configurar o tamanho dos buffers (Recomendação da Cloudera para o ambiente);
17. Configurar InnoDB (Formato de armazenamento banco de dados mais performático e recomendado pelo fabricante);
18. Iniciar e habilitar serviços do banco de dados MySQL (Inicializar o MySQL);
19. Definir a senha de administração para o MySQL (Segurança);
20. Criar bancos de dados MySQL e seus usuários;
21. Configurar replicação no Master Node (Redundância nas configurações e metadados do ambiente);
22. Configurar replicação nos Slave Nodes (Redundância para múltiplo acesso e persistência dos dados, assim como sua segurança);

23. Implantar conector MySQL (Necessário para que os componentes do Hadoop possam se conectar no MySQL);
24. Instalar repositório Cloudera (Instalação da aplicação);
25. Instalar pacotes do Cloudera Manager Server Node (Instalação da aplicação);
26. Preparar CDH para banco de dados externo (Preparação de Schema para abrigar os dados que virão do Cloudera Manager);
27. Iniciar serviço Cloudera Manager ServerStart Cloudera Manager Server;
28. Instalar pacotes Worker Nodes (Distribuir o Hadoop nos outros nós);
29. Instalar os componentes descritos no item 2.1 através da interface gráfica.

2.5 INSTALAÇÃO DO APACHE SPOT

Os seguintes passos foram utilizados para a instalação dos componentes necessários para o Apache Spot:

1. Criação de usuário (privilégios de super user):
2. Adição do usuário criado para o supergrupo HDFS;
3. Baixar o código-fonte do “spot-setup” e “spot-ingest” da Internet e colocar no diretório “/home” do servidor;
4. Ir ao módulo de configuração do Spot e editar as configurações de variáveis (“spot.conf”) de acordo com a documentação oficial;
5. Copiar o arquivo de configuração anterior para o diretório “/etc/”;
6. Executar o script “hdfs_setup.sh” para as diferentes funcionalidades (Netflow, DNS e Proxy) para criar um banco de dados no Hive e executar scripts de query hive necessários para acessar os dados;

2.6 INSTALAÇÃO DOS COMPONENTES DA FERRAMENTA

Os passos a seguir foram seguidos para a instalação dos componentes da ferramenta:

2.6.1 SPOT INGEST

1. Criar um diretório “src” para instalar todas as dependências do “spot-ingest”;
2. Instalar o gerenciador de pacotes “pip – python”;
3. Instalação do cliente Python do sistema de processamento em streaming distribuído Apache Kafka (“kafka-python”);
4. Instalação da biblioteca de APIs Python e os utilitários shell para monitoração dos eventos do sistema de arquivos (“watchdog”);
5. Instalação da ferramenta de dissecação de Netflow com características específicas para o “spot-ml” (“spot-nfdump”);
6. Instalação da ferramenta de dissecação de DNS (“tshark”);
7. Instalação do utilitário “screen”;
8. Download do arquivo “spark-streaming-kafka-0-8-assembly_2.11.jar” para suportar o Spark Streaming e o Kafka;
9. Configurar componente Ingest com as especificações do ambiente.

2.6.2 MACHINE LEARNING

1. Copiar código ML para o nó primário que iniciará a aplicação Spark;
2. Criar diretório src para instalar todas as dependências do código
3. Instalar aplicação “sbt—“ para construir códigos escritos em Scala;
4. Construir aplicação Spark que será o motor generalista para processamento de dados em larga escala;
5. Garantir que as configurações (“spot.conf”) estão no diretório “/etc/spot.conf”, de forma que serão replicadas para o restante dos nós

2.6.3 OPERATIONAL ANALYTICS

1. Copiar código do Operational Analytics para o servidor que será instalado no ambiente;
2. Adicionar arquivos de contexto no diretório “spot-ao/context_folder” que conterão o contexto de rede e geolocalização para a aplicação;

3. Adicionar arquivo de IP range (É utilizado pelo Operational Analytics quando está analisando arquivos de Netflow. Ele contém a lista de ranges de IP e o seu nome especificado);
4. Adicionar arquivo de localização de IP's (Ele contém um arquivo CSV com os ranges de IP no formato "integer" e as coordenadas para cada um desses ranges;
5. Adicionar arquivo de contexto de redes (Arquivo utilizado pelo Operational Analytics nos dados de DNS e Proxy, contendo o IP e seu respectivo nome);
6. Instalar executável "spot-setup" (Contém o script para instalar o banco de dados do Hive e inclui as principais configurações para o Operational Analytics, contendo diferentes variáveis que o usuário pode customizar de acordo com a instalação);
7. Conferência de configurações no arquivo "spot.conf" ("Luser" – diretório Home, "Huser" - diretório do HDFS, "Impala_Dem" – nó que o Impala Deamon está em execução, "DBNAME" – Banco de dados Hive);
8. Configuração de módulos Python inclusos no projeto que darão contexto e detalhes para os dados que serão analisados. Os componentes não são estritamente necessários, mas é recomendado para configurar todos eles em caso novos tipos de dados sejam analisados no futuro:
    a. Módulo de fonte de dados;
    b. Módulo de checagem de reputação;
    c. Módulo de tradução de Autoridade de Números Atribuídos na Internet;
    d. Módulo de contexto de rede;
    e. Módulo de Geolocalização;
9. Atualizar o arquivo "engine.json" com as informações do nome do nó que o Hive ou Impala está instalado e configurado e o nome do nó em que o serviço de banco de dados está em execução;
10. IPython com módulo de notebook habilitado que habilitará mudanças no código do projeto em tempo real;
11. Verificar se os arquivos da interface gráfica do Spot estão disponíveis;
12. Instalar visualização ("browserfy" e "uglify") como comandos globais no sistema;
13. Instalar dependências e construir a interface gráfica do Spot;

2.7 INGESTÃO DE DADOS

De forma a realizar a ingestão de dados, o fluxo necessário acontece de uma forma similar para as três fontes de dados (NetFlow, DNS e Proxy). Seguindo a documentação disponibilizada pela Apache, cada uma acontece da seguinte maneira: Os dados são colocados em um diretório no disco do servidor (coletor) e então através das ferramentas nfdump, tshark e parser (para NetFlow, DNS e Proxy, respectivamente) e então são colocados num tópico do Kafka no cluster e processados em um dos Worker Nodes, exceto os logs de Proxy, que utiliza o Spark Streaming, para que sejam estruturados no Hive, dentro do HDFS. As informações são colocadas numa tabela externa dentro do HDFS e, desta forma, não é possível realizar o processo de atualização no Hive, sendo necessário apagar a tabela e cria-la de novo.

O fluxo pode ser representado seguindo a figura 14 a seguir:

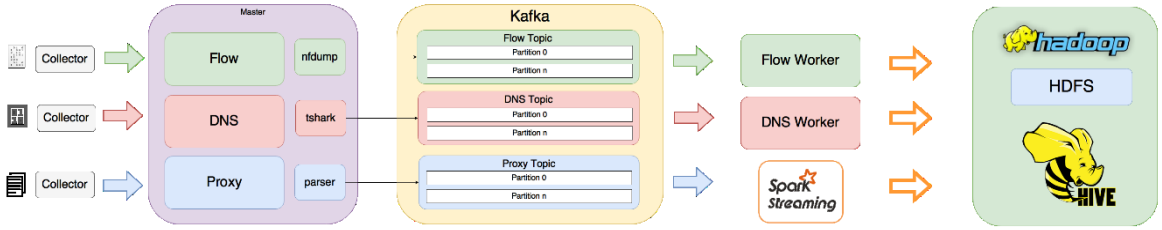


Figura 14: Fluxo de Ingestão de Dados [Site Oficial Apache Spot - 2018]

Os dados utilizados para a análise de NetFlow e Proxy não conseguiram ser ingeridos por diversos erros no processo utilizado e indicado pelo guia do usuário documentado.

Para o processo de ingestão do conjunto de dados de DNS, o seguinte comando foi utilizado:

```
./start_ingest_standalone.sh dns 2
```

Onde “2” é o número de Worker Nodes que realizaram todo o processamento. Após todo a ingestão, os dados podem ser avaliados na interface gráfica da ferramenta.

O conjunto de dados utilizado possuía 1 hora de duração de captura de dados e 6 milhões de registros divididos em 21 arquivos CSV, utilizando o Wireshark 2.0.4 e filtrado apenas para incluir repostas DNS. Para o ensaio sugerido, um único ataque que emula o tunelamento de uma troca HTTP de requisição/resposta utilizando o protocolo TCP sobre

DNS. O payload foi codificado nas requisições DNS utilizando a codificação “hex0x20Hack”, sendo capaz de extrair informações ou inserir um novo código dentro da requisição DNS.

Este ataque foi simulado em meio a um tráfego regular com 78535 clientes fazendo consultas de DNS em 8 servidores aleatoriamente.

2.7.1 DESCRIÇÃO DE COLUNAS

A tabela abaixo descreve o arquivo que foi ingerido para o ambiente de análise:

Cabeçalho da Coluna	*Descrição*
Frame.time	Timestamp do registro
Frame.time_epoch	Timestamp do registro em formato Epoch
Frame.len	Volume de captura do registro
Ip.src	Endereço IP da fonte
IP.dst	Endereço IP destino
Dns.resp.name	Busca DNS e nome de resposta
Dns.resp.type	Busca DNS e tipo de resposta
Dns.resp.class	Busca DNS e classe de resposta
Dns.flags.rcode	Código de resposta DNS
Dns.a	Endereço DNS

Tabela 01: Descrito de Colunas Ingeridas

Todos os dados referenciados acima na tabela 01 foram analisados através do Wireshark e filtrados para a análise dentro do Apache Spot.

3 RESULTADOS E DISCUSSÕES

Após a conclusão de todos os passos descritos na metodologia, a instalação dos servidores foi realizada com sucesso utilizando a AWS e, a ferramenta foi implantada de acordo com todas as especificações da documentação e os ajustes necessários para o ambiente criado (de forma reduzida à arquitetura de referência).

Devido à falta de maturidade da ferramenta, foram constatados diversos problemas de ingestão e na interface gráfica, mas também mostrando o potencial da ferramenta. As figuras abaixo demonstram a tela inicial quando solicitada a interface gráfica via browser, conforme apresentado na figura 15:

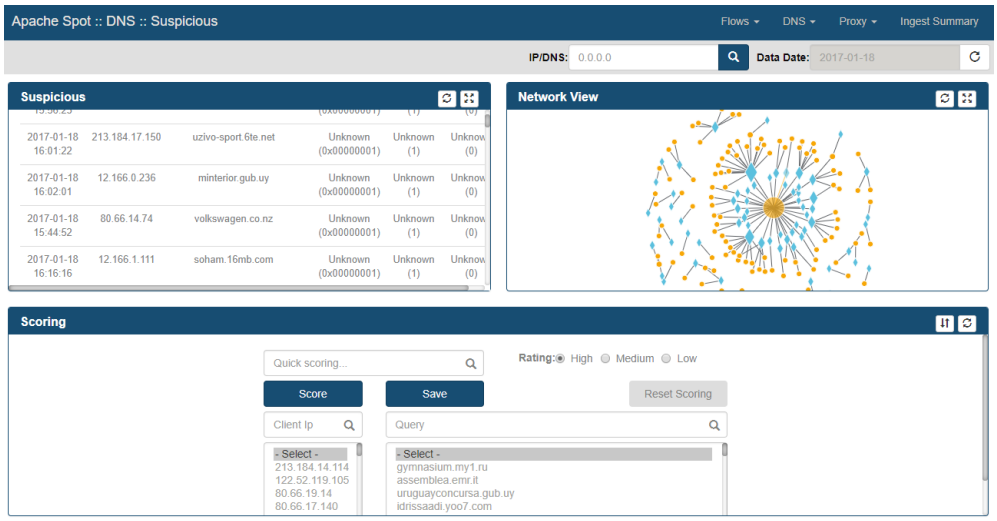


Figura 15: Interface gráfica de análise de DNS

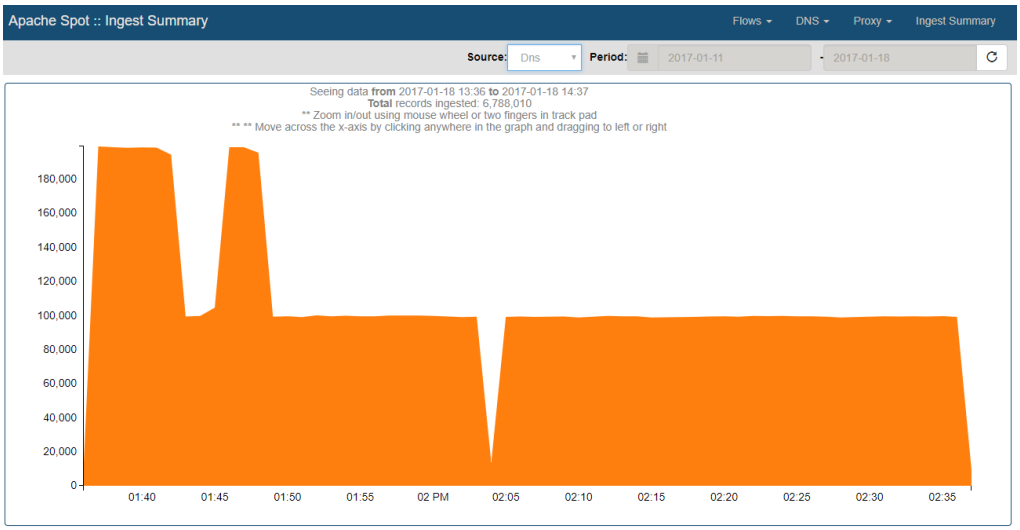


Figura 16: Interface de sumário de ingestão realizada

Como apresentado na metodologia, os conjuntos de dados de Proxy e NetFlow não foram ingeridos corretamente, e não foi possível a utilização dos algoritmos de aprendizado de máquina para analisar e buscar ameaças e ataques.

Na figura 16, é possível observarmos a quantidade de arquivos ingeridos no período selecionado para os logs de DNS.

Com o ensaio com os logs de DNS realizado, foi possível verificar o seguinte resultado na interface gráfica:

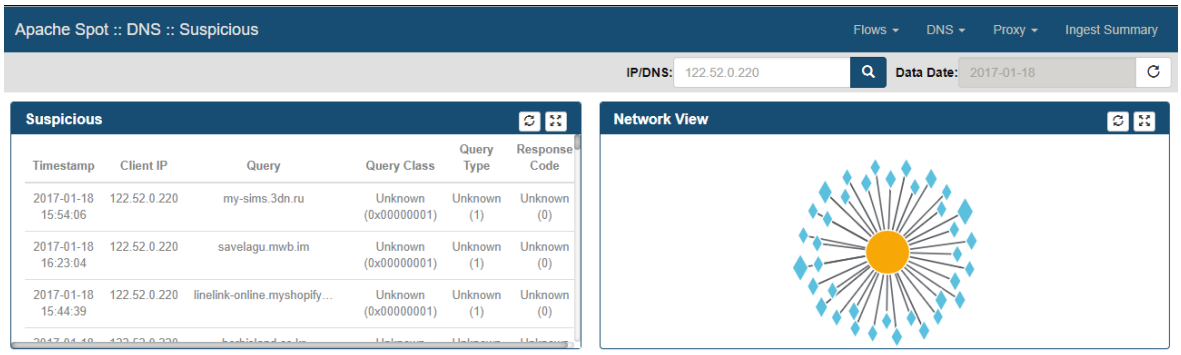


Figura 19: Interface gráfica de análise de DNS resultante

Conforme apresentado pela documentação do projeto, o quadro “Suspicious” apresenta as 250 maiores possíveis ameaças para os logs selecionados, após processamento do algoritmo de aprendizado de máquina baseado em LDA (Latent Dirichlet Allocation).

Nos resultados demonstrados na figura 19, os resultados foram filtrados com o IP 122.52.0.220, mostrando que, dentre os 6 milhões de registros, o algoritmo conseguiu apresentar os resultados esperados como ameaças a serem tratadas pela equipe de segurança. Através do grafo representado na ferramenta, os resultados são demonstrados de uma forma gráfica, em que os losangos representam registros de DNS e os círculos representam endereços IP se comunicando com o respectivo registro DNS.

Análise Suspeita apresentada pelo Apache Spot:

- **Servidor DNS:** 25.0.0.3
- **Requisição:** 122.52.96.104
- **Resposta:** xn--100-083bniwd8jlbt45a6a05rsgz588by24cmkbq51bth8an3r9f7g.com A 49.52.46.49 A 49.52.46.49 A 49.52.46.49 A 49.52.46

Analisando através o Wireshark, temos que um dos exemplos apresentados anteriormente simula um tunelamento de ruído aleatório utilizando TCP sobre o DNS, apresentando uma carga codificada na requisição DNS utilizando hex0x20Hack. É possível observar que a consulta de DNS não é apenas uma consulta comum, conforme esperado pelo conjunto de dados analisado.

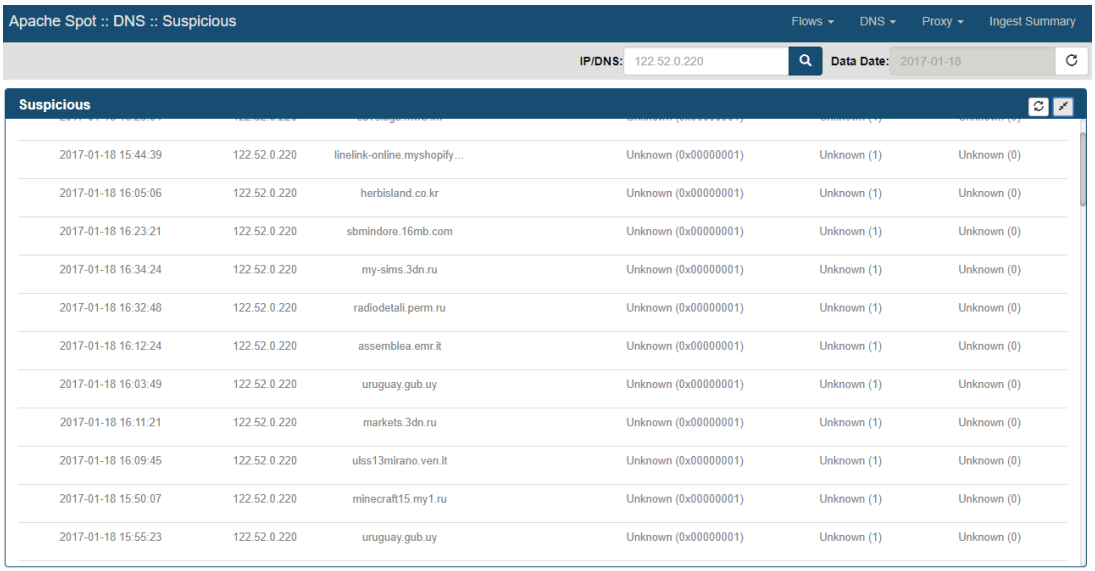


Figura 16: Resultados das análises suspeitas de DNS

Mais uma vez, mostrando algumas falhas na ferramenta, que precisa ser finamente ajustada para receber diversos conjuntos de dados e se tornar um verdadeiro “Datalake”, capaz de trabalhar com volumes imensos de dados de quaisquer fontes e trazer análises responsivas e assertivas sobre a comunicação de grandes redes.

Outra característica interessante da ferramenta é que, ela possibilita um chamado “Quick Scoring” para ranquear possíveis ameaças. Desta forma, é possível melhorar o algoritmo de aprendizado de máquina (LDA), trazendo melhores resultados, visto que existem alguns endereços já detectados como ameaças para a rede e, quando for executado de novo, o algoritmo já notará essas marcações. O problema apresentado por este “Quick Scoring” é que, os dados são replicados milhares de vezes para que o algoritmo identifique a ameaça no futuro previamente, não sendo uma boa prática o dado replicado desta maneira.

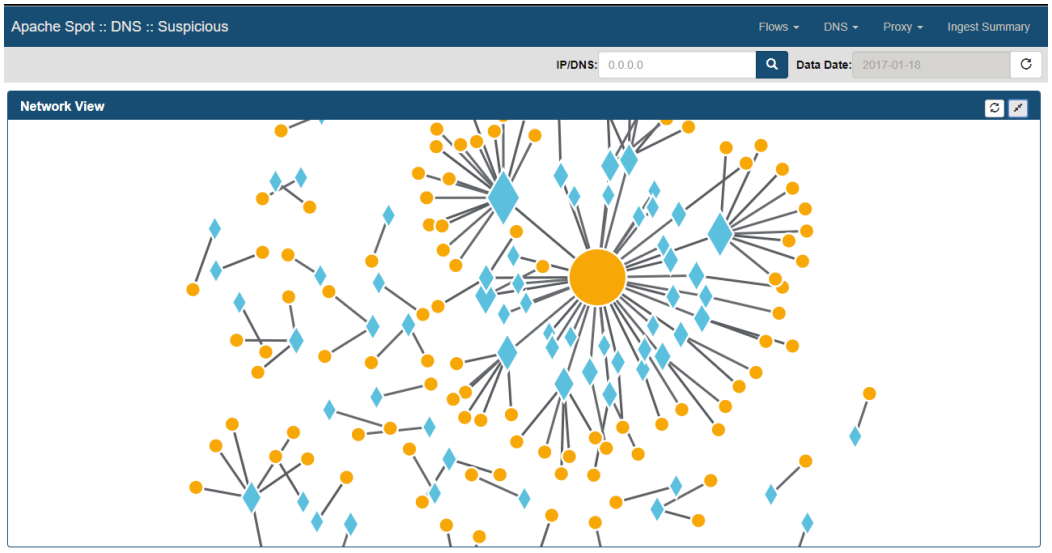


Figura 20: Grafo demonstrativo de todas as análises suspeitas

Através da visualização de rede que a ferramenta apresenta, é possível verificar uma visão diferenciada para os analistas de segurança, de forma a facilitar rastreamento e os maiores pontos de falha, conforme ilustrado na figura 20.

Após a utilização da característica de pontuação de IP's com alto risco, é possível verificar no menu "Threat Investigation" todos endereços salvos como alto risco serão apresentados e as respectivas consultas nos servidores DNS. Desta forma, ele pode ser salvo para que seja tomada alguma ação posterior e comentada pelo analista de segurança.

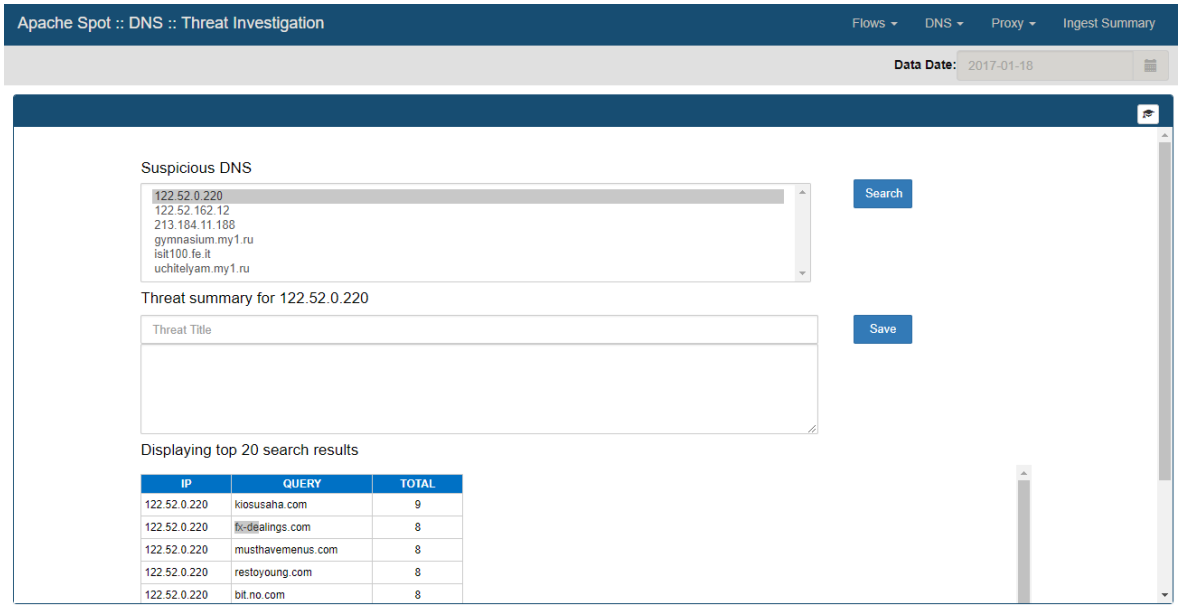


Figura 21: Ameaças de Segurança

Finalmente, através do Storyboard, as ameaças investigadas, são armazenadas e todas as consultadas realizadas pelo determinado IP ficarão guardadas, identificando toda a progressão do incidente determinado, conforme mostrado na figura 22.

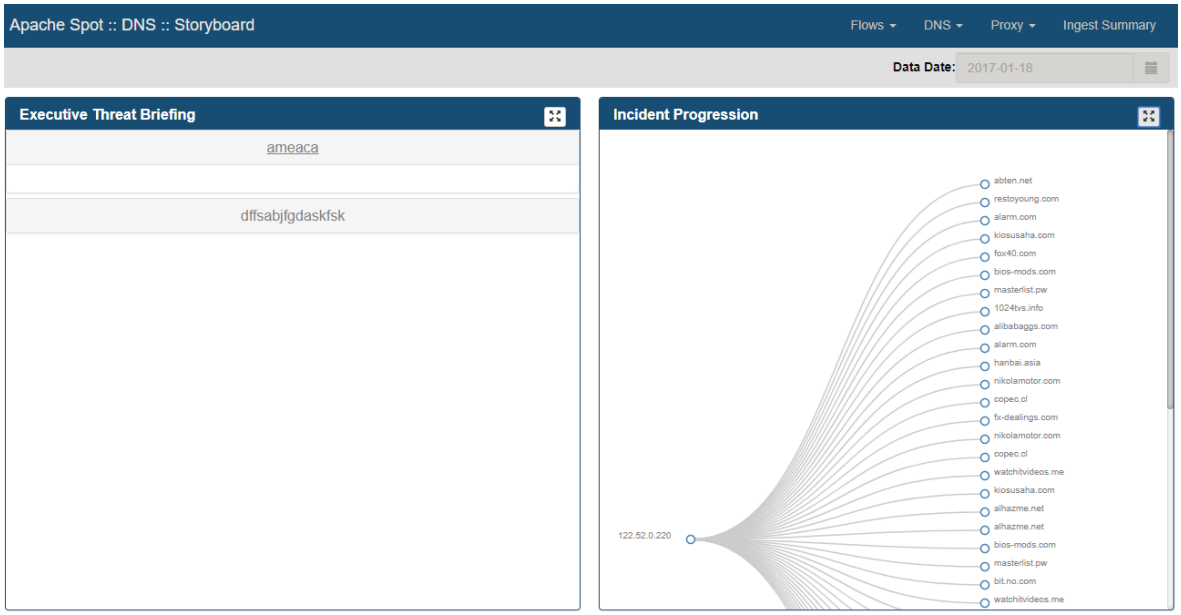


Figura 22: Progressão dos incidentes.

Devido à fase de desenvolvimento e por ser um projeto de código aberto, existe o modo avançado, que o código do projeto está detalhado e editável para os usuários:

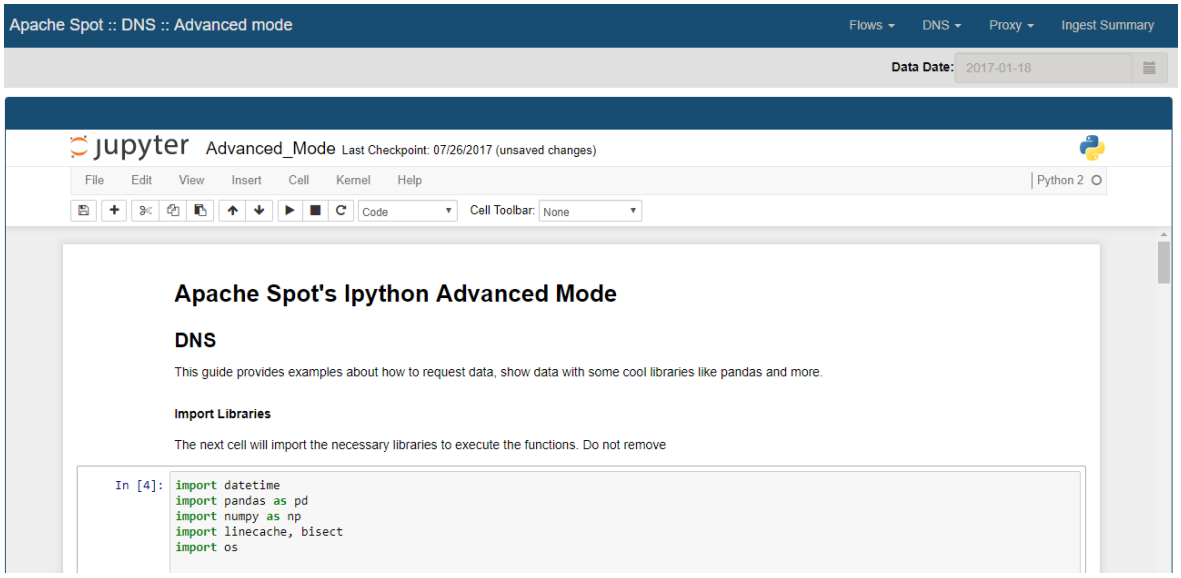


Figura 23: Modo avançado para melhorias no código

Possivelmente não é muito interessante a existência deste modo avançado para mudanças no código do programa, devido à possíveis mudanças estruturais que podem inviabilizar ou prejudicar o funcionamento total da ferramenta de uma forma muito simples.

4 CONCLUSÃO

Através do trabalho de graduação apresentado neste documento, foi possível analisar quase que completamente as funcionalidades da ferramenta Apache Spot (originalmente batizada como Open Network Insight), assim como diversos aspectos da área de segurança da informação que está constantemente presente no mundo atual e cada dia mais necessita de aprimoramentos, tendo em vista o futuro da área de tecnologia e todas as vulnerabilidades que são exploradas por pessoas mal-intencionadas.

A expansão crescente do número de equipamentos e servidores torna a área de segurança cada dia mais complexa, sendo necessário controlar e analisar escalas enormes de eventos todos os dias para garantir ambientes seguros para tráfego de dados. Desta forma, surge a necessidade de ferramentas extremamente escaláveis para grandes volumes de dados e com certa inteligência para analisar essas informações, pois seria impossível para analistas verificarem todo o conteúdo e encontrar possíveis ameaças aos seus sistemas.

Ferramentas como o ArcSight (HP), Splunk Enterprise (Splunk), QRadar (IBM), LogRhythm são SIEMs (Security Information and Event Management) que possuem uma característica semelhante ao que foi apresentado pelo Apache Spot, de armazenar altos volumes de dados de segurança e analisar eventos para detectar possíveis ameaças.

Entretanto, o diferencial competitivo encontrado neste momento foi a qualidade de ser gratuito e extremamente escalável, potencial que não se encontra nessas ferramentas, por serem muito caras (e.g.: Licenças ArcSight custando em torno de US$ 365.000,00 para 1000 eventos por segundo) e inviáveis para pequenos projetos ou negócios.

Ainda é muito cedo para que a ferramenta apresente sucesso, visto que está sendo desenvolvida e melhorada apenas pela comunidade. Muitos problemas foram encontrados com os datasets que foram testados durante a execução do projeto (Conjunto de dados de Proxy e Netflow não foram ingeridos com sucesso e sem mensagens de erro indicando o problema), e a interface ainda não está intuitiva e amigável para o usuário. Não há uma tela de login para o usuário (sem tratativa de segurança no ambiente) e o código está exposto para eventuais mudanças dos seus usuários.

Outra característica extremamente relevante encontrada foi a utilização de um Open Data Model, que é capaz de trazer e armazenar diversos tipos de informação de tráfego de redes diferentes para o mesmo ambiente para análise.

De forma geral, é possível concluir que a ferramenta Apache Spot tem um alto potencial para o mercado futuro, sendo capaz de substituir grandes ferramentas de mercado pelo simples fato de ser gratuita e escalável. Apesar de imatura, o projeto é promissor no que diz respeito às melhorias em sua inteligência e formato de apresentação e uso pelas equipes de Segurança da Informação e substituição de grandes jogadores de mercado por conta dos altos preços.

5 REFERÊNCIAS

1. Documentação Oficial Apache Spot (Acessado em 22/08/2017)
2. Repositório de código aberto do Apache Spot (Acessado em 22/08/2017)
3. COLOURIS, G.; DOLLIMORE, J.; KINDBERG, *T. Sistemas Distribuídos – Conceitos e Projeto.* 4ºEd. Porto Alegre: Bookman, 2007;
4. BHAJI Y. *CCIE Professional Development – Network Security Technologies and Solutions,* 1º Ed., Indianapolis: Cisco Press, 2008.
5. STALLINGS W. *Network Security Essentials – Applications and Standards*, 4ºEd., Boston, Pearson, 2011.
6. NAKAMURA E.; GEUS, P. *Segurança de Redes em Ambientes Cooperativos*, São Paulo, Novatec, 2010.
7. International Organization for Standardization – Informações sobre a ISO 27001 - (Acessado em 07/12/2018)
8. M.BLEI, DAVID; Y. NG ANDREW; I. JORDAN, MICHAEL *Latent Dirichlet Allocation*, 2003, Research Paper, Stanford University, Palo Alto, 2003
9. BELL, DAVID ELLIOTT *Look Back at the Bell-La Padula Model,* 2005, ResearchPaper, Annual Computer Security Applications Conference, Tucson, 2005
10. BALON, NATHAN; THABET, ISHRAQ *Biba Security Model Comparison,* 2004, Research Paper, University of Michigan, Michigan, 2004
11. BLAKE, SONYA Q *The Clark-Wilson Security Model,* 2000, Research Paper, SANS INSTITUTE, Bethesda, 2000
12. MCLEAN, JOHN *Security Models and Information Flow,* 1990, Research Paper, IEE Security & Privacy, Washington, 1990
13. GUPTA, VARUN *Chinese Wall Security Policy,* 2009, Tese de Mestrado, San Jose State University, San Jose, 2009
14. SANDHU, RAVI S. *Lattice-Based Access Control Models*, 1993, Research Paper, George Mason University, Fairfax, 1993
15. SIMONEAU, PAUL *The Osi Model: Understanding the Seven Layers of Computer Networks, 2006, Research Paper, University of Patras, Rio, 2006*
16. CISCO SYSTEMS, INC *Network Security Baseline, San Jose, 2000*
17. Documentação Oficial Cloudera Enterprise Data Hub (Acessado em 22/08/2017)