

Universidade de Aveiro

Segurança em Redes de Comunicações

Relatório do Projeto 2



universidade de aveiro

André Clérigo (98485), Pedro Rocha (98256)

Departamento de Eletrônica, Telecomunicações e Informática

11 de junho de 2023

Conteúdo

| | |
|---|----|
| 1. Introdução | 3 |
| 2 Metodologia | 4 |
| 2.1 Análise de Dados | 4 |
| 3 Análise de comportamento não anômalo | 5 |
| 3.1 Comunicações internas | 5 |
| 3.2 Comunicações Externas | 6 |
| 4 Detecção de comportamento anômalo | 8 |
| 4.1 Atividade de botnet | 8 |
| 4.2 Ataques C&C | 9 |
| 4.3 Exfiltração de Dados | 10 |
| 4.4 Comunicações Suspeitas de Países | 11 |
| 5 Definição de regras SIEM | 13 |
| 5.1 Aumento do Acesso ao Servidor | 13 |
| 5.1.1 Teste de Regras e Identificação de Dispositivos | 13 |
| 5.2 Comunicações Internas | 14 |
| 5.2.1 Teste de Regras e Identificação de Dispositivos | 15 |
| 5.3 Países impertinentes. | 15 |
| 5.3.1 Teste de Regras e Identificação de Dispositivos | 16 |
| 5.4 Exfiltração de Dados | 16 |
| 5.4.1 Teste de Regras e Identificação de Dispositivos | 17 |
| 5.5 Novos Protocolos | 17 |
| 5.5.1 Teste de Regras e Identificação de Dispositivos | 18 |
| 5.6 Novas Portas | 18 |
| 5.6.1 Teste de Regras e Identificação de Dispositivos | 19 |
| 6. Conclusão | 20 |

Lista de Figuras

| | | |
|-----|--|----|
| 2.1 | Uso médio dos protocolos. | 4 |
| 3.1 | IPs de Servidor Potenciais. | 5 |
| 3.2 | Média de fluxos para redes externas e internas. | 6 |
| 3.3 | Os 5 principais países com mais fluxos. | 7 |
| 3.4 | Os 5 principais países com mais bytes carregados. | 7 |
| 3.5 | Os 5 principais países com os bytes mais baixados. | 7 |
| 4.1 | Comportamento de Botnets. | 8 |
| 4.2 | Aumento dos fluxos dos IPs com mais fluxos no conjunto de dados de teste. ... | 9 |
| 4.3 | Aumento dos fluxos do conjunto de dados normal para o conjunto de dados de teste. | 10 |
| 4.4 | Média de bytes carregados. | 11 |
| 4.5 | Estatísticas dos Países. | 12 |
| 5.1 | Resultado da regra de aumento de acesso ao servidor. | 14 |
| 5.2 | Resultado da Regra de Comunicação Interna. | 15 |
| 5.3 | Resultado da regra dos países impertinentes. | 16 |
| 5.4 | Resultado da Regra de Exfiltração de Dados. | 17 |
| 5.5 | Resultado da regra de novos protocolos. | 18 |
| 5.6 | Resultado da Nova Regra de Portas. | 19 |

Capítulo 1

Introdução

À medida que a tecnologia digital se torna uma parte fundamental nos negócios atuais, também assistimos a um aumento nos riscos potenciais de segurança cibernética. À medida que as empresas continuam a aumentar a sua pegada digital, é da maior importância ter sistemas fortes e fiáveis em funcionamento. Pensando nisso, o objetivo do projeto é identificar comportamentos incomuns e dispositivos potencialmente comprometidos utilizando sistemas de gerenciamento de informações e eventos de segurança (SIEM).

O objetivo principal deste projeto é definir e implementar regras SIEM que possam identificar efetivamente atividades de rede incomuns que possam indicar ameaças potenciais. O projeto utiliza dados históricos de fluxos de tráfego de uma rede “corporativa”, sabendo que um dos conjuntos de dados fornece um comportamento típico da rede e o outro conjunto de dados contém comportamentos de rede anômalos.

Com base na análise destes conjuntos de dados, o resultado do projeto será um conjunto de regras SIEM, concebidas a partir da nossa análise, que podem identificar ameaças potenciais, quer sinalizando-as como um alarme, quer bloqueando ações com base na gravidade da ameaça.

Concluindo, o projeto é um exercício de segurança de rede, utilizando análise de dados para compreender o comportamento da rede, definir regras de alerta e testar sua eficácia. Este esforço não é apenas um exercício de segurança cibernética, mas um passo crucial na salvaguarda da integridade da rede corporativa.

Capítulo 2

Metodologia

O projeto funciona de forma metódica, começando com a análise do conjunto de dados "dataX.parquet", que contém dados de um dia inteiro sobre o comportamento típico da rede. Esses dados, já certificados como livres de comportamento ilícito, constituem a base para a nossa compreensão das operações normais da rede.

Posteriormente, nos aprofundamos no conjunto de dados "testX.parquet". Este conjunto de dados, espelhando a estrutura do conjunto de dados anterior, mas contendo potencialmente comportamentos anômalos, nos permite contrastar atividades de rede normais e anormais e identificar padrões únicos que sinalizam ameaças potenciais.

A análise de dados utilizará Python, aproveitando a biblioteca pandas para análise de dados. Paralelamente, utilizaremos bancos de dados para geolocalização baseados em endereços IPv4, também utilizamos Jupyter Notebooks para uma análise direta dos dados o que simplificou bastante a criação das regras SIEM.

2.1 Análise de Dados

Na parte de análise de dados do projeto, começamos coletando informações simples dos conjuntos de dados usando pandas tanto dos conjuntos de dados normais quanto de teste como as portas e protocolos utilizados, avaliando se os utilizados foram: UDP na porta 53, UDP na porta 443 e TCP pela porta 443 que representam DNS, QUIC e HTTPS, respectivamente. É possível verificar qual a média de utilização dos protocolos a partir da Figura 2.1.

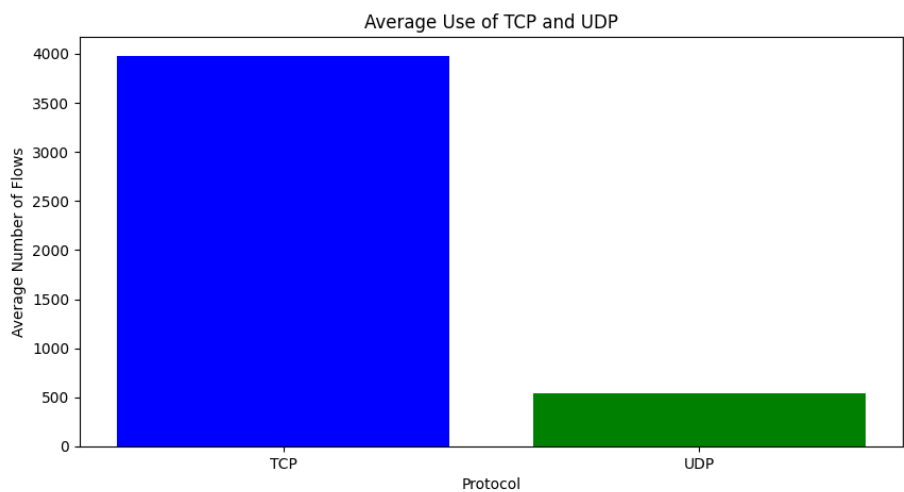


Figura 2.1: Média de utilização dos protocolos.

Analizamos também a quantidade de fluxos por IP de origem, bytes carregados e bytes baixados por fluxo e total, tanto para comunicações internas (IP privado para IP privado) quanto para comunicações externas (de IP privado para IP público). Coletando também métricas sobre valor médio, desvio padrão e variância. Adicionalmente, foram também recolhidas algumas estatísticas de países, tendo em conta todas as métricas anteriormente referidas.

Capítulo 3

Comportamento Não Anômalo

Análise

Discuta o comportamento típico dos dispositivos de rede com base nos dados analisados. Tendo este conjunto de dados normal como ponto de partida para qual é o comportamento típico da rede quando não há nenhuma atividade maliciosa, coletamos algumas informações dele para posteriormente compará-las com o conjunto de dados de teste que possui padrões de comunicação anômalos.

3.1 Comunicações internas

Ao explorar as comunicações internas (entre endereços IP privados), encontramos vários comportamentos normais como a lista de endereços IP que normalmente utilizam a rede e suas comunicações dentro dela. Sobre os protocolos e portas utilizadas, observou-se que não há conexões TCP na porta 53. Esta porta geralmente é utilizada para tráfego DNS. Esta descoberta sugere que o tráfego DNS nesta rede está usando UDP em vez de TCP, o que é típico para consultas e respostas DNS. Além disso, observou-se que um número significativo de conexões UDP utilizava a porta 443, que normalmente está associada ao tráfego HTTPS utilizando TCP. O uso de UDP nesta porta sugere que o protocolo QUIC pode estar em uso. QUIC é frequentemente usado para melhorar a velocidade de carregamento de páginas da web e reduzir a latência no tráfego da web. Ainda nesta parte analisamos o número de fluxos UDP e TCP para cada IP de origem, onde descobrimos quais IPs eram os pertencentes aos Servidores da rede. Como é possível verificar na Figura 3.1, os IPs com mais fluxos são os endereços potenciais dos servidores, então ficamos com os IPs dos servidores: 192.168.101.234, 192.168.101.239, 192.168.101.224 e 192.168.101.228. O outro IP (192.168.101.154) possui uma quantidade de tráfego mais regular.

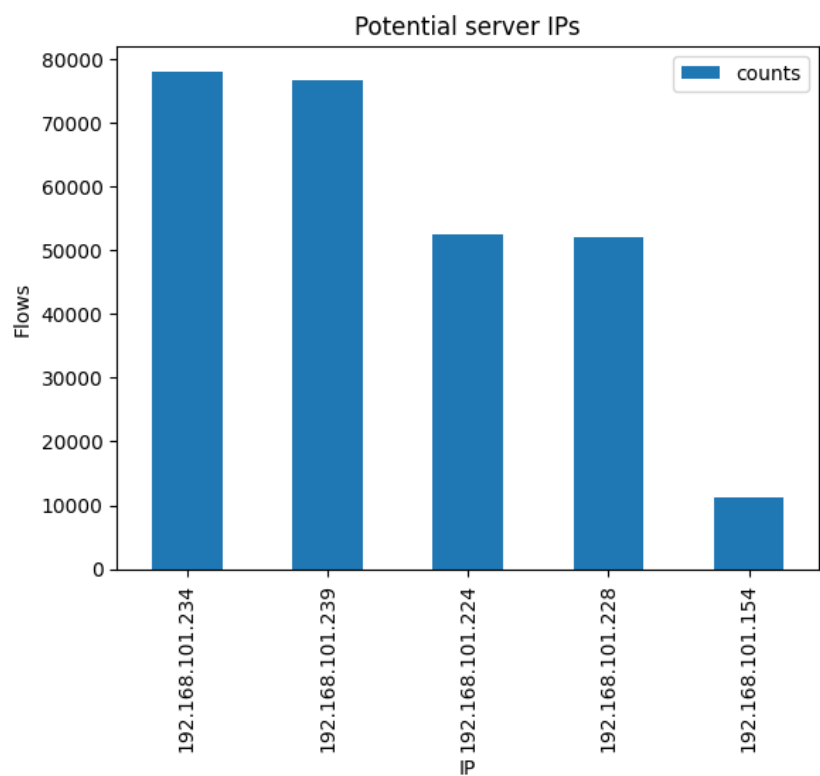


Figura 3.1: IPs de servidores potenciais.

Na figura acima podemos ver os 5 principais IPs privados com mais fluxos. É claro que o 5º IP não possui tantos fluxos quanto os demais, mas isso por si só não significa que este IP não seja um servidor. Podemos avaliar que este IP não é um servidor porque sua contagem de fluxo (11162) está alinhada com os valores médios da contagem de fluxo (4518).

3.2 Comunicações Externas

Nesta seção adotamos a mesma abordagem de antes, analisando quais protocolos, portas, endereços IP e densidade de fluxo existiam. No entanto, também conseguimos recolher quais os países e organizações que estavam a ser contactados a partir da rede interna e quanto isso estava a acontecer, para que tivéssemos uma base de quais eram as comunicações internas para a rede externa. Obtendo informações relevantes como a mostrada na Figura 3.2. Também obtivemos valores como fluxos por país, fluxos por organização, etc.

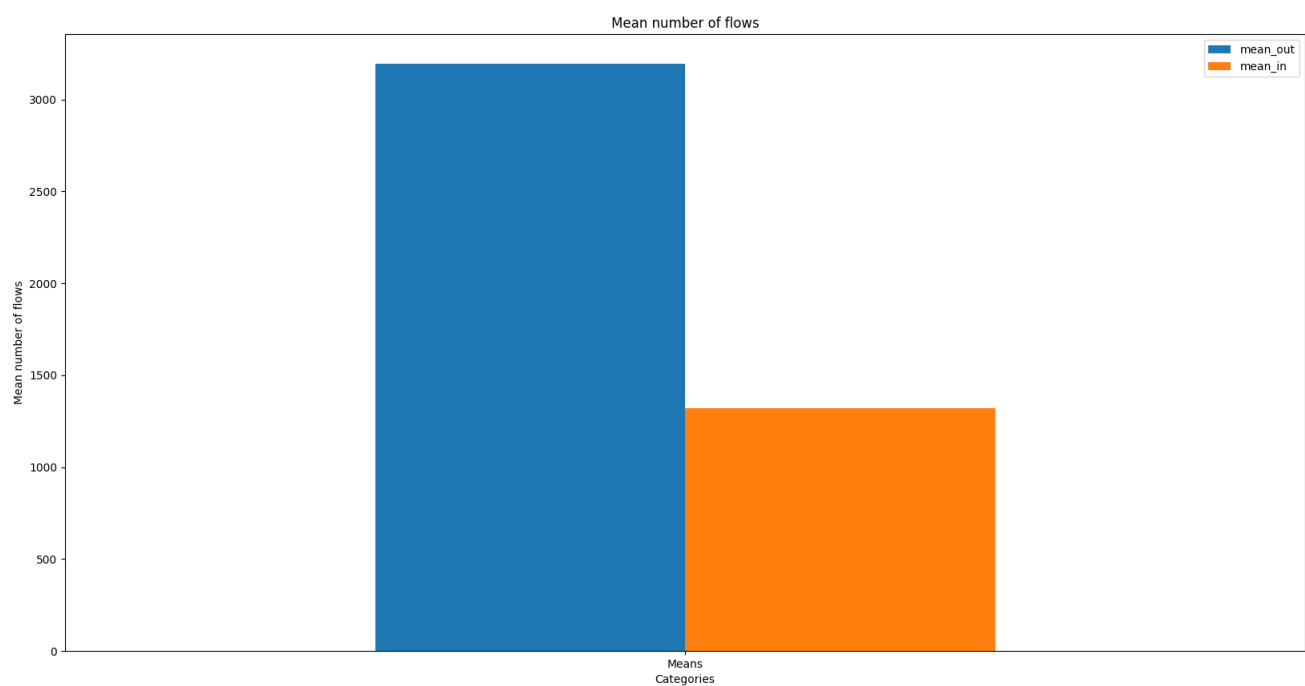


Figura 3.2: Média de fluxos para redes externas e internas.

Para referência, analisamos os 5 principais países com mais fluxos, bytes carregados e bytes baixados. Os países com mais fluxos, bytes carregados e bytes baixados obtidos foram Estados Unidos da América, Portugal, Holanda, Namíbia e Grã-Bretanha respetivamente.

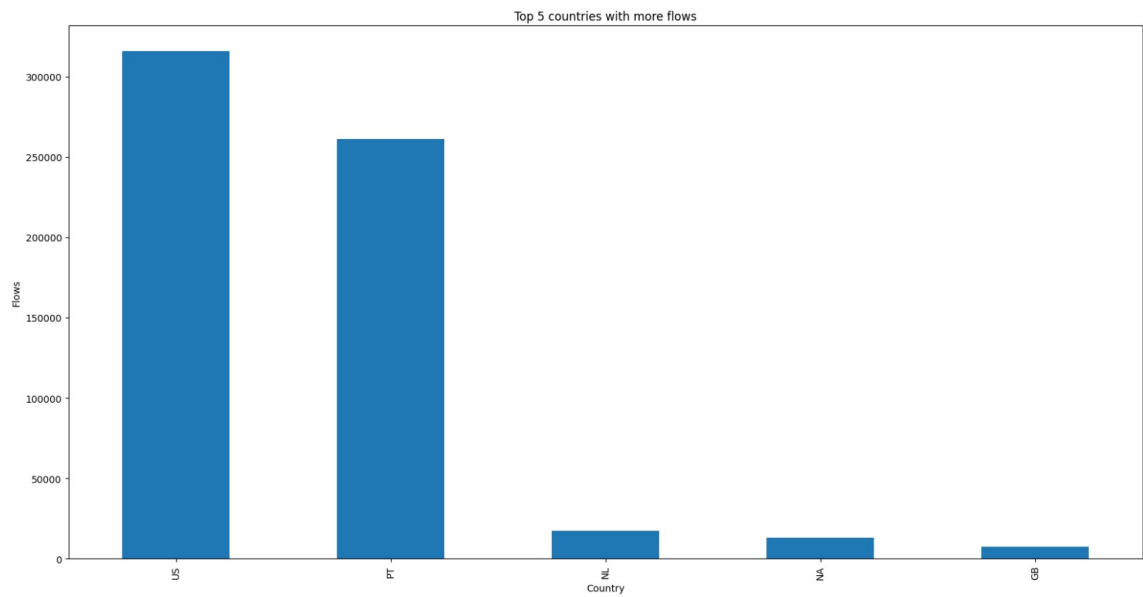


Figura 3.3: Os 5 principais países com mais fluxos.

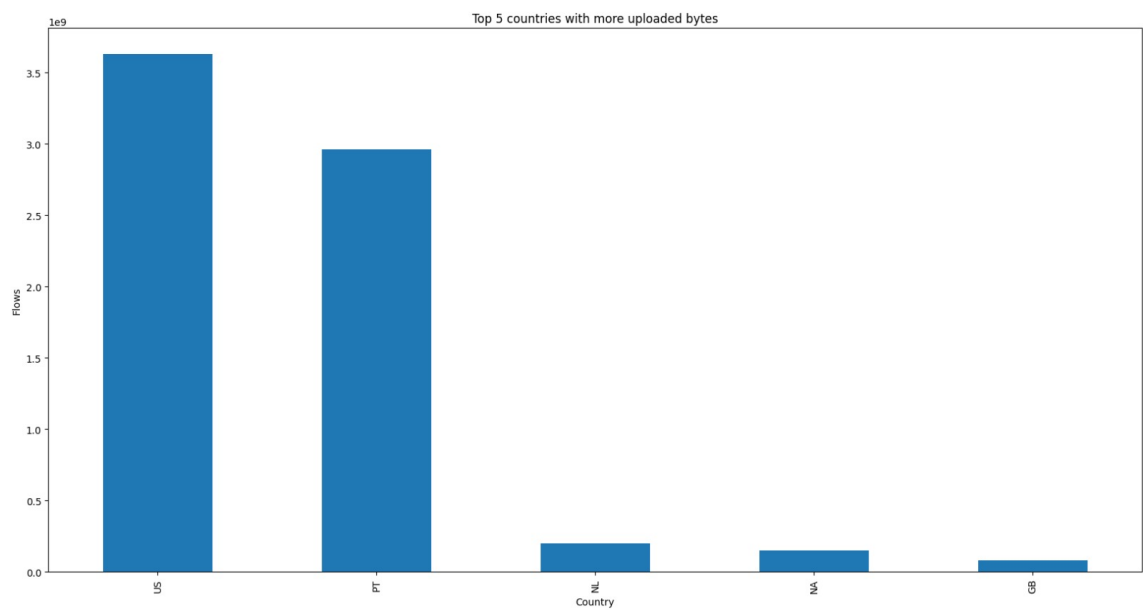


Figura 3.4: Os 5 principais países com mais bytes carregados.

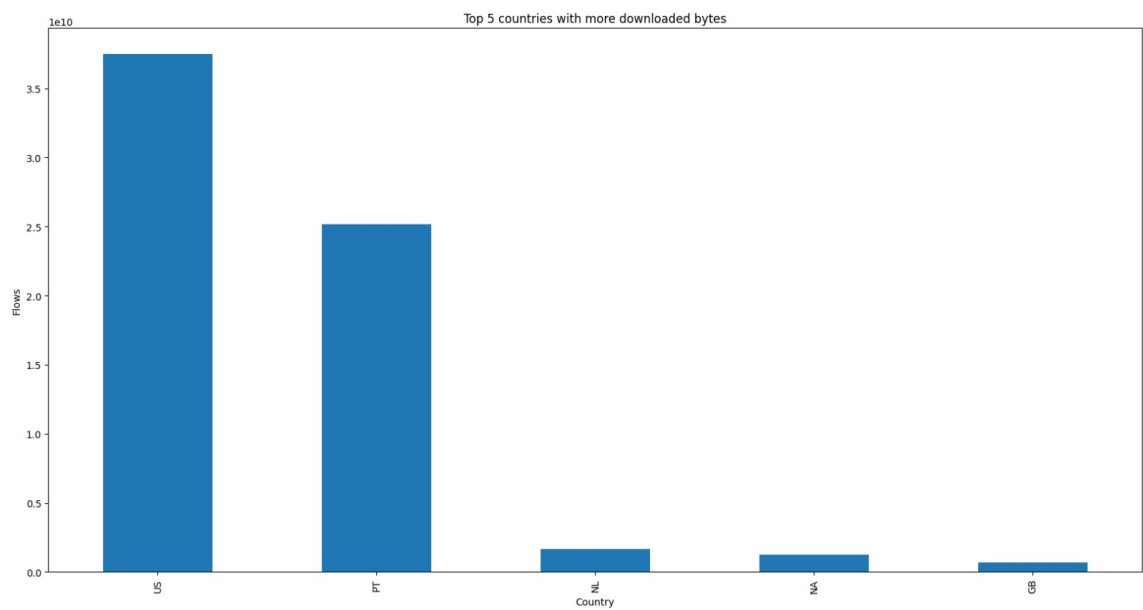


Figura 3.5: Os 5 principais países com os bytes mais baixados.

Capítulo 4

Detecção de comportamento anômalo

4.1 Atividade de botnet

Uma botnet é um conjunto de dispositivos comprometidos que estão sob o controle de um comando central. Esses dispositivos comprometidos, muitas vezes chamados de “bots”, são normalmente infectados com software malicioso sem o conhecimento de seus proprietários.

Com base nisso, pegamos os endereços IP que estão envolvidos em novas comunicações internas e os consideramos suspeitos. A seguir, analisamos se suas comunicações são normais ou maliciosas, observando fatores como volume anormalmente alto de dados, presença de numerosos fluxos, etc.

Os IPs que detectamos como suspeitos foram 192.168.101.11, 192.168.101.110, 192.168.101.14 e 192.168.101.34.

Para determinar se esses endereços IP fazem parte de uma botnet, uma análise mais aprofundada deve ser realizada.

Como exemplo podemos referir a Figura 4.1 que mostra um novo IP fazendo conexões com alguns dos servidores, porém, essas conexões possuem uma pequena janela de tempo para que um humano faça essas muitas solicitações. Este tipo de análise é algo que não conseguimos realizar com automação e tivemos que analisar “manualmente”. Os outros tipos de anormalidades são fáceis de detectar com uma regra.

Na nossa investigação, identificamos vários endereços IP envolvidos em comunicações internas suspeitas. Embora algumas anormalidades possam ser facilmente detectadas com regras automatizadas, a identificação de padrões de conexão incomuns pode exigir análise manual. O isolamento fornece tempo para essa análise, permitindo que as equipes de segurança reúnam mais informações.

Botnet Suspect IP 192.168.101.34 communicating internally with other possible botnets.

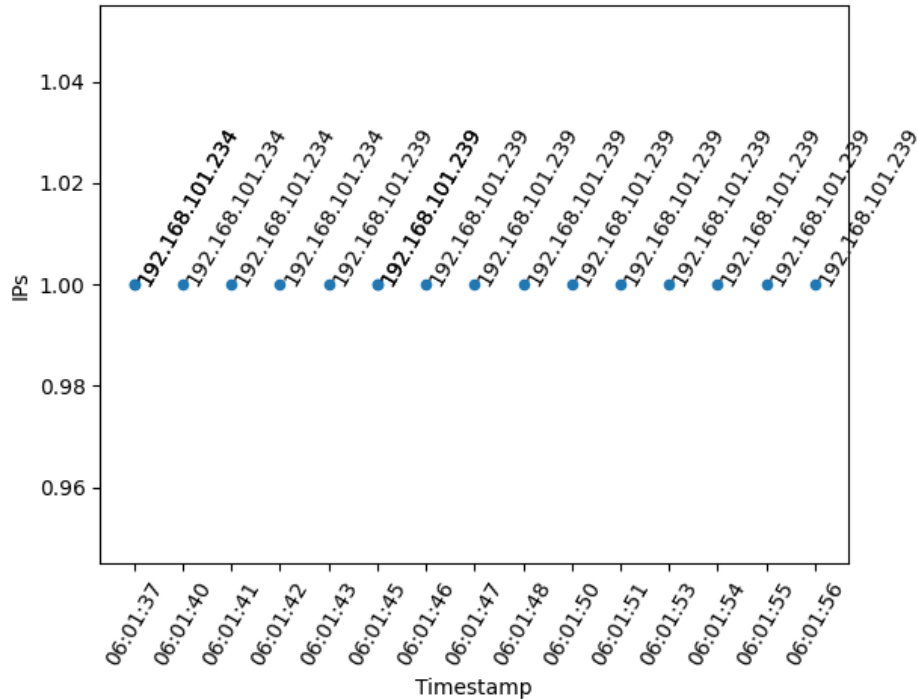


Figura 4.1: Comportamento do Botnet.

4.2 Ataques C&C

Na secção dedicada à avaliação do Comando e Controlo (C&C) do nosso projeto, é crucial destacar os eventos relevantes que ocorreram durante a fase de monitorização. Através de uma análise meticulosa dos dados, percebeu-se que os servidores estavam sob um ataque notável. Isto foi inferido a partir de um aumento incomum no número de fluxos direcionados aos IPs do servidor (224 e 228), como é possível ver na Figura 4.2. Normalmente, o tráfego de rede, ou “fluxos”, permanece dentro de um determinado intervalo esperado. No entanto, durante o período de avaliação, foi observado um aumento considerável nos fluxos direcionados aos IPs dos servidores, o que foi indicativo de uma possível atividade de C&C.

Além disso, a nossa investigação aprofundada revelou que houve um aumento abrupto na comunicação entre determinados endereços IP e os servidores. Isto manifestou-se num aumento substancial dos fluxos destes IPs (41, 42, 60) para os servidores como é possível ver na Figura 4.3. Embora sejam esperadas algumas flutuações no tráfego durante as operações normais, o grande volume e a rapidez deste aumento no número de fluxos levantaram preocupações. É importante mencionar que determinados endereços IP apresentam um aumento significativo no número de fluxos, ultrapassando os 1000%. No entanto, estas contagens de fluxo elevadas permanecem dentro da faixa do número médio de fluxos observados durante o tráfego diário regular.

Para quantificar o comportamento observado, o acesso médio ao servidor foi calculado e comparado com as contagens de fluxos de endereços IP individuais. Os endereços IP que apresentavam um número de fluxos significativamente superior à média foram reservados para um exame mais aprofundado. Especificamente, foram sinalizados IPs cujos fluxos eram superiores a cinco vezes a média de acesso ao servidor. Isto permitiu-nos restringir as fontes potenciais que poderiam fazer parte da rede C&C.

Num ataque C&C, os sistemas comprometidos comunicam frequentemente com um servidor de comando para receber instruções ou enviar dados. O aumento incomum nos fluxos de IPs específicos para os servidores pode implicar que esses sistemas estão comprometidos e fazem parte de uma botnet usada para o ataque C&C. Esse tipo de comportamento é sintomático de sistemas que foram comandados para executar tarefas sob o controle de um invasor remoto.

Concluindo, os IPs suspeitos devem ser isolados e minuciosamente analisados quanto a qualquer conteúdo ou comportamento malicioso. Além disso, o tráfego de rede deve ser monitorado continuamente em busca de quaisquer padrões incomuns, e devem ser implementadas medidas preventivas para mitigar o risco de futuros ataques C&C. Esta experiência sublinha a necessidade de mecanismos de segurança dinâmicos e robustos para proteger a integridade e a confidencialidade dos dados e sistemas.

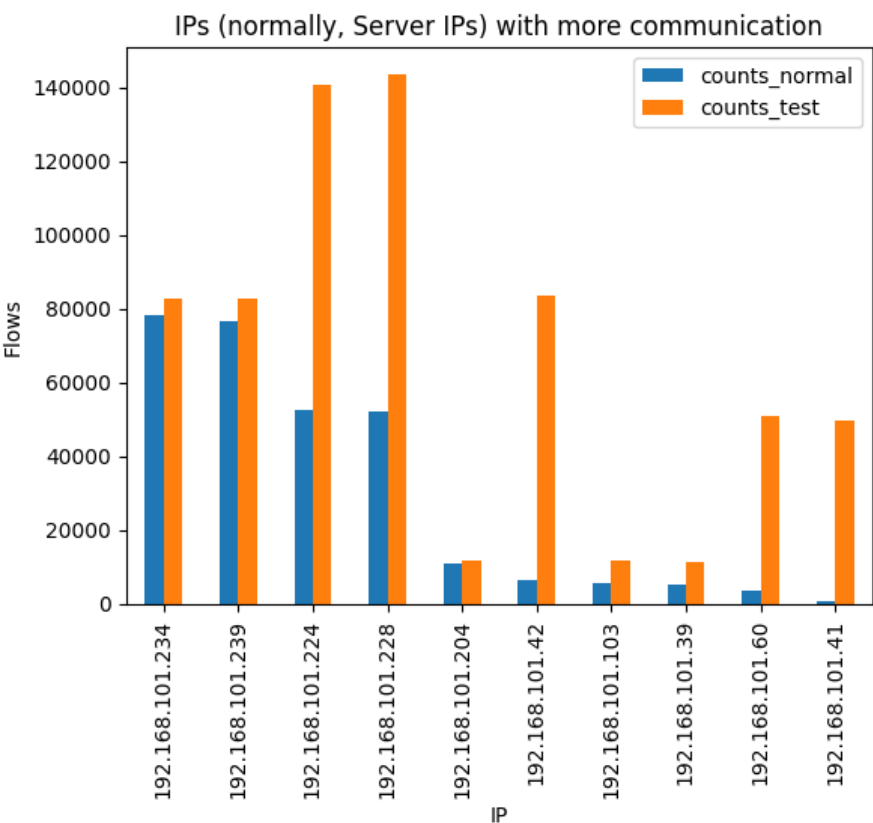


Figura 4.2: Aumento de fluxos dos IPs com mais fluxos no conjunto de dados de teste.

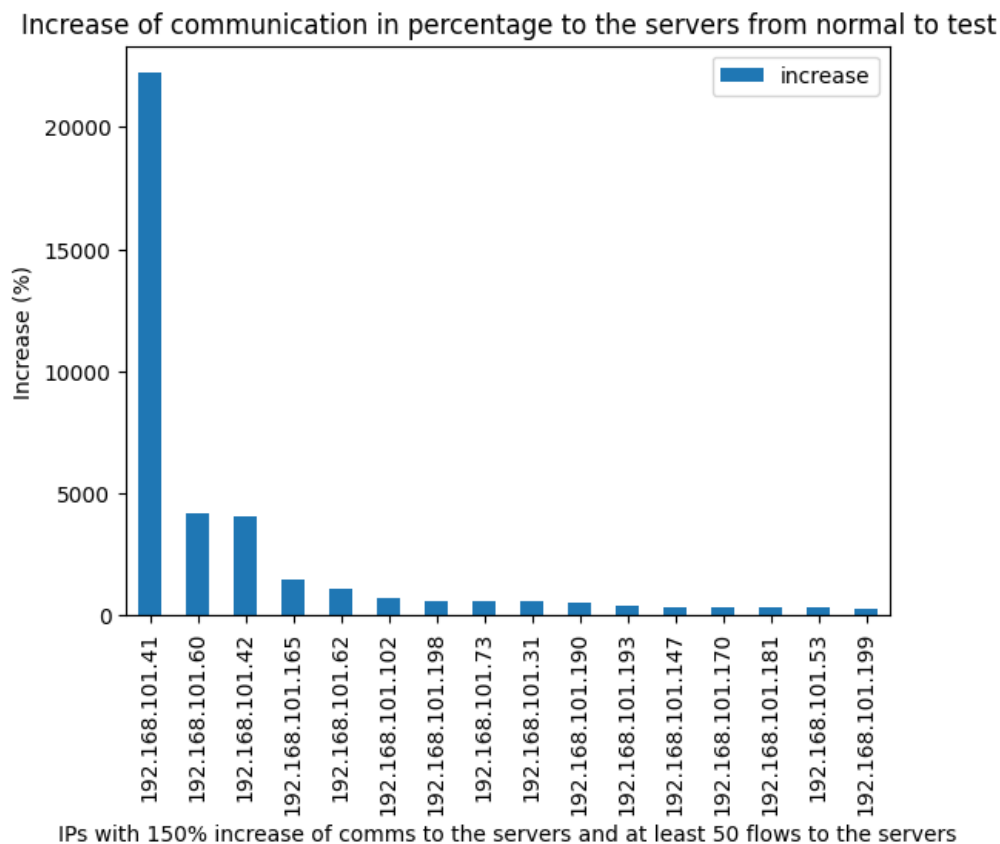


Figura 4.3: Aumento dos fluxos do conjunto de dados normal para o conjunto de dados de teste.

4.3 Exfiltração de Dados

Nesta seção de nossa análise, nos concentramos na identificação de possível exfiltração de dados e comunicações externas anômalas. A exfiltração de dados pode ser uma ameaça séria, pois envolve a transferência não autorizada de dados de dentro da organização para um local externo, comprometendo potencialmente informações confidenciais.

Inicialmente, a análise envolve a identificação de novos endereços IP que passaram a se comunicar externamente. O código separa esses endereços IP comparando a lista de IPs de origem nos dados atuais (teste de fluxos externos) com os IPs de origem dos dados de linha de base (fluxos externos). Este processo também é replicado para IPs de destino. Isso ajuda a identificar quaisquer padrões de comunicação novos ou incomuns com endereços IP externos que não estavam presentes antes e podem ser indicativos de uma violação de segurança ou de uma conexão não autorizada.

Uma vez identificados os novos endereços IP que se comunicam externamente, a análise se aprofunda investigando a quantidade de dados que esses IPs estão carregando e baixando. Isto é particularmente importante para identificar atividades de exfiltração de dados. Especificamente, a análise calcula o volume total de dados em megabytes carregados e baixados por esses IPs suspeitos. Um aumento repentino nos dados carregados pode ser um indicativo de exfiltração de dados da rede, como acontece na Figura 4.4.

Além disso, a análise também investiga os endereços IP públicos acessados por esses IPs suspeitos, bem como os países e organizações associados a esses IPs públicos.

Além disso, a análise envolve verificar o aumento percentual da comunicação com o exterior, fundindo os conjuntos de dados de base e de teste e calculando o aumento no número de fluxos. IPs com aumento superior a 150% e com mais de 50 fluxos são destinados a um exame mais detalhado.

Na subseção Exfiltração de dados, a ênfase está no volume de dados enviados para fontes externas. É importante reconhecer que a exfiltração de dados geralmente envolve um volume maior de dados carregados em comparação com dados baixados. A análise avalia a média de bytes carregados por fluxo e procura aumentos substanciais.

Além disso, o código procura padrões de comunicação periódicos que enviam uma quantidade consistente de dados, o que pode ser um sinal de alerta para processos automatizados de exfiltração.

No final, a análise envolve representação visual por meio da plotagem da média de bytes carregados por fluxo para os IPs envolvidos na exfiltração de dados. Isso auxilia na análise visual, facilitando a identificação de anomalias.

Concluindo, esta seção da análise é fundamental para identificar e compreender qualquer potencial exfiltração de dados ou atividades de comunicação externa não autorizadas. Através do exame detalhado de novos IPs, dos volumes de dados transferidos e da natureza da comunicação externa, os analistas de segurança podem tomar decisões informadas sobre possíveis ameaças à segurança e implementar medidas para proteger a rede.

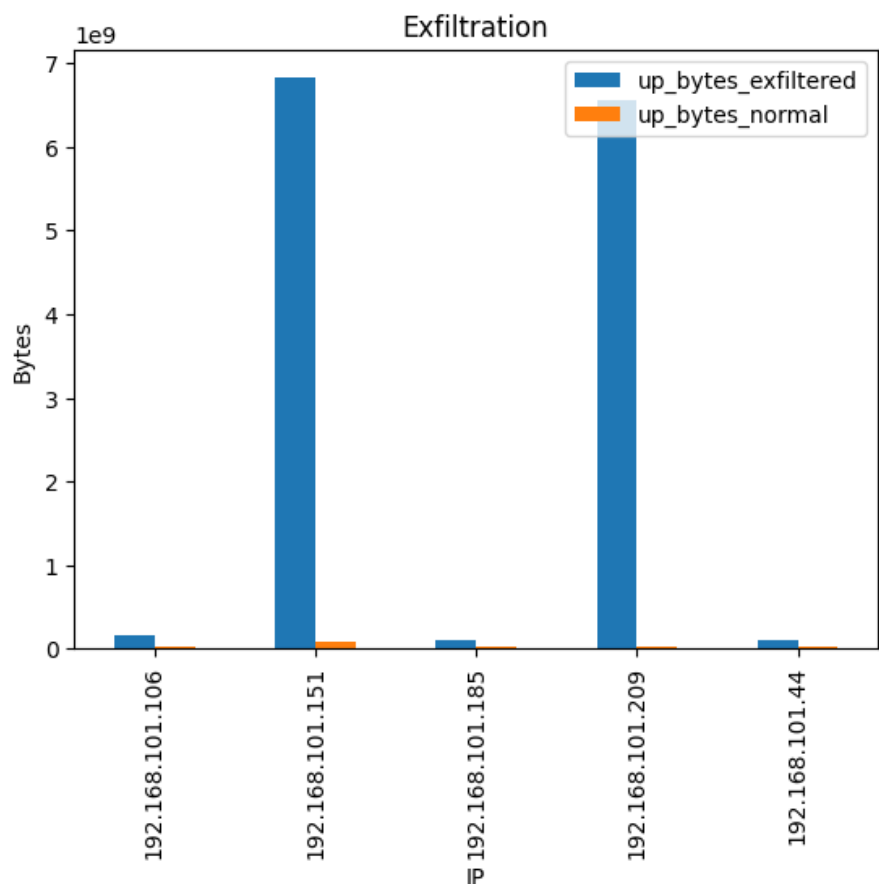


Figura 4.4: Média de bytes carregados.

4.4 Comunicações Suspeitas do País

Nesta seção do relatório, examinamos os dados de comunicação da rede para analisar as interações com países externos. Isto envolve compreender a distribuição de fluxos de rede, bytes carregados e bytes baixados para diferentes países e identificar quaisquer comunicações incomuns ou anômalas.

- **Agregação e análise de dados** - Primeiro, agregamos os dados por códigos de país de destino (dst cc) e calculamos as contagens totais de fluxos de rede, bem como a soma de bytes carregados (bytes superiores) e bytes baixados (bytes inferiores) para cada país . Para uma análise mais detalhada, calculamos também a média de bytes carregados e baixados para cada país. As comunicações locais, onde o endereço IP de destino começa com '192.168.101.', são excluídas desta análise.
- **Visualização dos principais países comunicantes** - Visualizamos os dados para obter insights sobre os padrões de comunicação com países externos:
 - Número de Fluxos: Criamos um gráfico de barras exibindo os 5 principais países com maior número de fluxos de comunicação. Este gráfico revela quais países têm comunicação mais frequente.
 - Bytes carregados: outro gráfico de barras é criado para mostrar os 5 principais países para onde a maioria dos dados é enviada.
 - Bytes baixados: por último, um gráfico de barras exibe os 5 principais países dos quais a maioria dos dados é recebida.Essas visualizações nos ajudam a compreender os padrões usuais de comunicação e a estabelecer uma linha de base para o comportamento normal.
- **Análise Comparativa com Dados de Teste** - Para detectar quaisquer alterações ou anomalias significativas, comparamos o conjunto de dados de teste com o conjunto de dados normal.

Em seguida, analisamos as mudanças no número de fluxos e bytes e sinalizamos países com aumentos incomuns na comunicação. Considera-se que um país tem comunicação incomum se atender aos seguintes critérios:

O número de fluxos é superior a 200 e Há um aumento de pelo menos 50% no número de fluxos ou Há um aumento de pelo menos 50% nos bytes carregados/ baixados.

Com base nas observações da Figura 4.5, detectámos um ligeiro comportamento anormal nas comunicações com a China. Além disso, considerando a quantidade significativa de tráfego para a Rússia no conjunto de dados anômalo, em comparação com a quase nenhuma comunicação com esse país no conjunto de dados normal, recomendamos bloquear a comunicação com a Rússia.

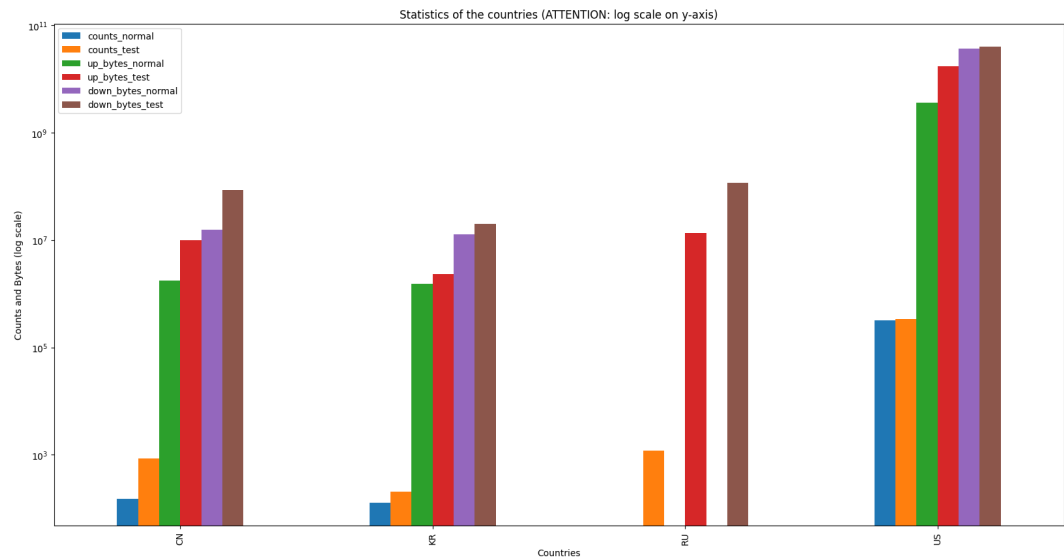


Figura 4.5: Estatísticas dos Países.

capítulo 5

Definição de regras SIEM

Uma regra SIEM é um conjunto predefinido de critérios no sistema SIEM. Ele foi projetado para detectar e responder a incidentes de segurança analisando dados de eventos de diversas fontes. As regras SIEM monitoram condições e padrões específicos para acionar alertas ou respostas automatizadas (como um bloqueio de comunicação). Ao usar regras SIEM, as organizações podem identificar e mitigar proativamente ameaças à segurança.

5.1 Aumento do acesso ao servidor

A regra a seguir visa resolver o problema de acesso ao servidor, que também detectará ataques de negação de serviço distribuída (DDoS).

A primeira etapa da regra envolve o cálculo do número médio de acessos a IPs de servidores específicos, nomeadamente 234, 239, 224 e 228, com base no conjunto de dados fornecido. Esta média serve como base ou ponto de referência para avaliar padrões de acesso subsequentes.

Para identificar potenciais eventos desencadeadores de alarme, a regra considera qualquer contagem de acesso que exceda três vezes a média calculada. Este limite indica um aumento significativo no acesso ao servidor, o que pode merecer mais atenção.

Além disso, a regra recomenda uma resposta mais rigorosa quando se descobre que um endereço IP tem uma contagem de acesso cinco vezes maior que a média. Este nível de atividade é considerado uma ameaça mais grave, indicando uma intenção potencialmente maliciosa. Nesses casos, a regra aconselha o bloqueio do endereço IP para evitar novos acessos não autorizados e mitigar o risco de um potencial ataque DDoS.

É importante observar que o trecho de código fornecido realiza apenas cálculos e identifica o acesso ao servidor de teste que excede os limites definidos. O código não inclui a implementação real de alarmes ou bloqueio de IP. Seria necessário integrar estas ações num sistema SIEM ou incorporá-las num código subsequente para operacionalizar totalmente a regra e responder eficazmente a potenciais ameaças DDoS.

Concluindo, esta regra verifica o aumento de acessos e conexões aos servidores e dispara um alarme caso haja IPs que atinjam mais de 3 vezes a média dos fluxos de volume e os bloqueia quando estiver 5 vezes acima dela.

```
1  # Média de acesso aos IPs do servidor (IPs do servidor: 234, 239, 224, 228)
2  acesso médio ao servidor = data.loc[(data['dstip'].isin(ips do servidor['índice']))] . agrupar por ([
3  'srcip']).size().reset índice(nome='conta')
4
5  acesso médio ao servidor = acesso médio ao servidor['conta'].mean() print("Média de acesso
6  de IPs do servidor: ln" + str(int(média de acesso ao servidor)))
7
8  # verifica os fluxos de teste que acessam os IPs do servidor (IPs do servidor: 234,...
9  239, 224, 228) com mais de 50 fluxos e 3 vezes a média
10 acesso ao servidor de teste = test.loc[(test['dstip'].isin(ips do servidor['índice']))] . agrupar por ([
11 'srcip']).size().reset índice(nome='conta')
12
13 acesso ao servidor de teste = acesso ao servidor de teste[acesso ao servidor de teste['conta']...
14 > (acesso médio ao servidor*3)]
15 print("Testar acesso aos IPs do servidor: ln" + str(teste de acesso ao servidor))
```

5.1.1 Teste de regras e identificação de dispositivos

Ao testar a regra, obtivemos os seguintes resultados:

A eficácia da regra é evidente porque detectou e alarmou com sucesso determinados IPs, incluindo aqueles associados a atividades maliciosas. Além disso, provavelmente sinalizou IPs que exibiam comportamento não anômalo, mas que justificavam uma investigação mais aprofundada. A regra provou ser valiosa no bloqueio dos IPs que claramente abusavam do acesso ao servidor, garantindo

```
Average server IPs access:
1322
The IPs that activate an alarm:
      src_ip  counts
192.168.101.139    4108
192.168.101.41    46161
192.168.101.42    77974
192.168.101.60    47980
The IPs to be blocked:
      src_ip  counts
192.168.101.41    46161
192.168.101.42    77974
192.168.101.60    47980
```

Figura 5.1: Resultado da regra de aumento de acesso ao servidor.

que apenas entidades de alto risco foram bloqueadas, fornecendo alertas para IPs que exigiam escrutínio adicional.

5.2 Comunicações Internas

A regra fornecida concentra-se no monitoramento e na resposta às comunicações internas dentro de uma rede, visando principalmente novas comunicações internas que possam exigir bloqueio. A regra segue uma série de etapas:

Em primeiro lugar, identifica as comunicações internas entre endereços IP privados no conjunto de dados normal e armazena as informações no dataframe interno normal. Em seguida, ele calcula a contagem média de comunicações internas tomando a média das contagens de comunicação no dataframe interno normal. Esta média serve como base para avaliar as comunicações internas subsequentes.

A regra então verifica o conjunto de dados de teste para identificar novas comunicações internas entre endereços IP privados. Ao filtrar os dados de teste e agrupá-los com base nos endereços IP de origem e destino, a regra calcula o tamanho de cada grupo de comunicação, armazenando as informações no dataframe interno de teste.

Para determinar novas comunicações internas no conjunto de dados de teste, a regra compara as comunicações internas do conjunto de dados normal (interno normal) com aquelas no conjunto de dados de teste (interno de teste). Ele seleciona e armazena as comunicações que existem apenas no conjunto de dados de teste, indicando novas comunicações internas, no dataframe diff interno.

Além disso, a regra verifica se alguma comunicação interna antiga do conjunto de dados normal tem contagens de comunicação superiores a três vezes a contagem média (média interna). Ele compara as comunicações internas antigas com as novas comunicações internas no conjunto de dados de teste e seleciona as comunicações antigas que ultrapassam o limite. Essas comunicações também são armazenadas no dataframe diff interno.

Finalmente, a regra gera as comunicações internas que dispararam um alarme. As informações incluem o endereço IP de origem, o endereço IP de destino e a contagem de comunicações para cada comunicação interna no dataframe de comparação interna.

Concluindo, não são permitidas novas ligações internas (seriam bloqueadas como no eduroam da UA) e as que foram permitidas não podem passar 3 vezes a média dos fluxos de acesso ao servidor, caso contrário dispara um alarme.

```
1 # Verificar para comunicações internas em normal (IP privado para IP privado) normal
2 interno = data.loc[(data['srcip'].apply(lambda x: ... -
      endereço IP.endereço IP(x).é privado)) &...
      (dados['dstip'].apply(lambda x: ipaddress.ip address(x).is private))]) normal internal = normal
3 internal.groupby(['srcip',... -
      'dstip']).size().reset índice(nome='conta')
4 # print("Comunicações internas normais: ln" + str(normal interno))
5
6 # Contagem média de comunicações internas
7 média interna = normal interna['conta'].mean() print("Fluxos médios de
8 comunicação interna: ln" +...
      str(int(média interna)))
9
```

```
10 # Verificar para novas comunicações internas em teste (IP privado para IP privado) test
11 internal = test.loc[(test['srcip'].apply(lambda x: ...
    endereço IP.endereço IP(x).é privado)) &...
    (teste['dstip'].apply(lambda x: ipaddress.ip address(x).is private))] teste interno = teste
12 interno.groupby(['srcip',... -
    'dstip']).size().reset índice(nome='conta')
13 # print("Comunicações internas em teste: ln" + str(teste interno)) -
14
15 # Obtenha a diferença para verificar as novas comunicações internas em test
16 internal diff = pd.merge(normal internal, test interno,... -
    ligado=['srcip','dstip'], como='certo') diferença
17 interna = diferença interna.fillna(0)
18 diferença interna = diferença interna[(diferença interna['conta x'] == 0) &...
    (diferença interna['conta você']>0)]
19 diferença interna = diferença interna[['srcip','dstip','conta você']] diferença interna =
20 diferença interna.rename(colunas={'conta você':'conta'}) -
21 # print("Novos fluxos de comunicação interna: ln" + str(diferença interna)) -
22
23 # Verifique as antigas comunicações internas se há algum que tenha mais... de 3 vezes a
    média
24 diferença interna = pd.merge (normal interno, teste interno,... -
    ligado=['srcip','dstip'], como='certo') diferença
25 interna = diferença interna.fillna(0)
26 diferença interna = diferença interna[(diferença interna['conta x']>... -
    (média interna*3)) & (diferença interna['conta você']>0)] diferença interna = diferença
27 interna[['srcip','dstip','conta você']] diferença interna = diferença interna.rename(colunas={
28 'conta você':'conta'}) print("Alarme sobre este fluxo de comunicação interna: ln" +...
29
    string interna diff.to (index=False))
```

5.2.1 Teste de regras e identificação de dispositivos

Ao testar a regra, obtivemos os seguintes resultados:

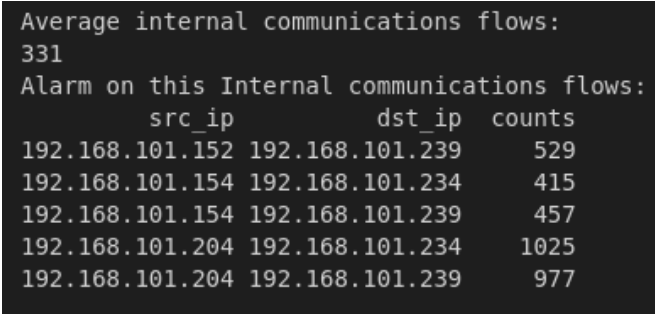


Figura 5.2: Resultado da Regra de Comunicação Interna.

Utilizando esta regra, conseguimos detectar duas categorias distintas de países: aqueles que registam um aumento substancial nos números de fluxos em comparação com a média, e aqueles que anteriormente não tinham histórico de comunicação, mas que agora iniciaram comunicações com um número invulgarmente elevado de fluxos (pelo menos menos 500 fluxos).

5.3 Países impertinentes

A seguinte regra analisa o número médio de fluxos por país agrupando os dados com base no código do país de destino (dst cc). Calcula a contagem média do fluxo entre países, proporcionando uma compreensão do nível de fluxo normal para cada país.

Em seguida, verifica o conjunto de dados de teste para países com contagens de fluxo superiores a 18 vezes a média ou com um número infinito de fluxos. Estes países são sinalizados como potenciais fontes de preocupação ou anomalias e é acionado um alarme.

A regra examina ainda mais o conjunto de dados de teste para identificar os países que existem no conjunto de dados de teste, mas não no conjunto de dados normal. Esses países são considerados novos ou anteriormente não observado no tráfego da rede.

Entre os países recentemente identificados, a regra seleciona aqueles com contagens de fluxo superiores a 500. Considera-se que estes países têm um volume de fluxos invulgarmente elevado e podem exigir bloqueio ou investigação adicional.


```
1 # Verifique a média de fluxos por país fluxos médios
2 país =... -
    dados.loc[~(dados['dstip']).str.startswith('192.168.101.')]
3 . agrupar por ([ 'dst cc' ]).size().reset índice(nome='conta') país de fluxos médios = país de
4 fluxos médios[ 'conta' ].mean() print("Fluxos médios por país: ln" + str(int(médio fluxos país)))
5
6
7 # Verificar se o teste tem mais de 18 vezes a média OU é INF de... fluxos por país

8 fluxos de teste país =...
    teste.loc[~(teste['dstip']).str.startswith('192.168.101.')]
9 . agrupar por ([ 'dst cc' ]).size().reset índice(nome='conta')
10 país dos fluxos de teste = país dos fluxos de teste[(país dos fluxos de teste[ 'conta' ]...
    > (média de fluxos por país*18))/(fluxos de teste país[ 'conta' ] ==... flutuador('inf'))]

11 print("Os IPs que ativam um alarme: ln" +...
    fluxos de teste country.to string(index=False))
12
13 # Verificar para países que existem em teste, mas não em fluxos normais de
14 teste país =... -
    teste.loc[~(teste['dstip']).str.startswith('192.168.101.')] . agrupar por ([ 'dst cc'
15 ]) .size().reset índice(nome='conta') -
16 país dos fluxos de teste = país dos fluxos de teste[~(fluxos de teste país[ 'dst cc' ] .
17 isin(dados.loc[(dados['dstip']).str.startswith('192.168.101.')]
18 . agrupar por ([ 'dst cc' ]).size().reset índice(nome='conta')[ 'dst cc' ]))] -
19
20 # Verifique quais desses países têm mais de 500 fluxos
21 país de fluxos de teste = país de fluxos de teste[país de fluxos de teste[ 'conta' ]...
    > 500]
22 print("Os IPs a serem bloqueados: ln" +...
    fluxos de teste country.to string(index=False))
```

5.3.1 Teste de regras e identificação de dispositivos

Ao testar a regra, obtivemos os seguintes resultados:

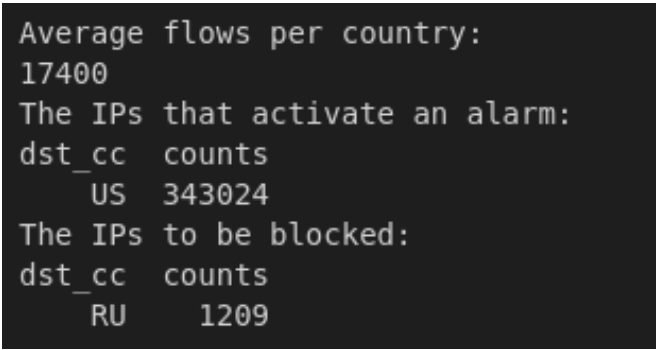


Figura 5.3: Resultado da regra dos países impertinentes.

A norma demonstrou sua eficácia ao gerar um alarme para o aumento dos fluxos observados nos EUA. Este aumento é provavelmente considerado normal devido ao papel do país como um centro para numerosos centros de dados e plataformas amplamente utilizadas. No entanto, a regra também identificou e bloqueou fluxos provenientes da Rússia, país que não tinha estado anteriormente envolvido em quaisquer comunicações.

5.4 Exfiltração de Dados

Esta regra de exfiltração de dados visa analisar a média de bytes de upload no conjunto de dados normal. Ele calcula o total de bytes de upload para cada endereço IP de origem (src ip), calcula a média de bytes de upload em todos os endereços IP e determina a média de bytes de upload.

Em seguida, a regra examina o conjunto de dados de teste para identificar fluxos que excedem três vezes a média de bytes de upload. Ele agrupa os dados pelo endereço IP de origem, calcula o total de bytes de upload para cada endereço IP e seleciona os fluxos que ultrapassam o limite.

Além disso, a regra verifica fluxos no conjunto de dados de teste que ultrapassam cinco vezes a média de bytes de upload. Segue um processo semelhante de agrupar os dados pelo endereço IP de origem e selecionar os fluxos que excedem o limite definido.

```
1 # Média de bytes de upload normalmente
2 média de bytes = data.groupby(['srcip'])['até bytes'].soma(). redefinir
3 índice(nome='até bytes')['até bytes'].mean().print("Média de bytes de upload: l
4 n" + str((média de bytes)))
5
6 # Verificar se existem fluxos em teste que passam 3 vezes a média... carregar bytes
7
8 testar_bytes = ...
9 teste.groupby(['srcip'])['até bytes'].sum().reset índice(nome='até bytes' testar bytes = testar
10 bytes[testar bytes['até bytes'] > ...
11 (média de bytes*3)]
12 print("Os IPs que ativam um alarme: ln" + ...
13 teste bytes.to string(index=False))
14
15 # Verificar se existem fluxos em teste que passam 5 vezes a média... carregar bytes
16
17 testar_bytes = ...
18 teste.groupby(['srcip'])['até bytes'].sum().reset índice(nome='até bytes' testar bytes = testar
19 bytes[testar bytes['até bytes'] > ...
20 (média de bytes*5)]
21 print("Os IPs a serem bloqueados: ln" + teste bytes.to string(index=False))
```

5.4.1 Teste de regras e identificação de dispositivos

Ao testar a regra, obtivemos os seguintes resultados:

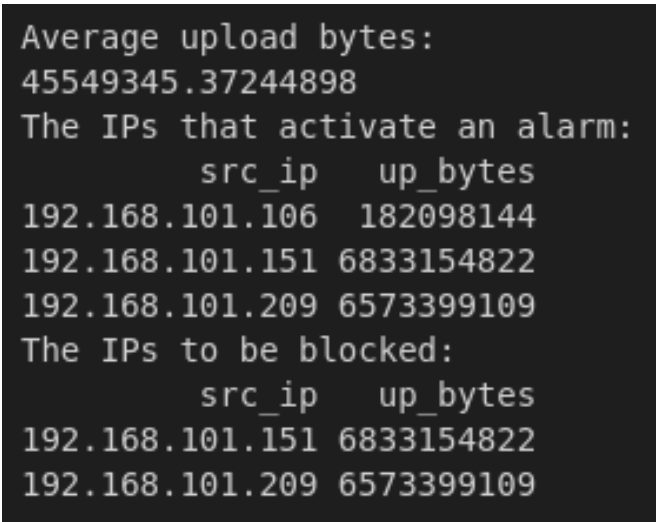


Figura 5.4: Resultado da regra de exfiltração de dados.

A regra se mostrou eficaz ao disparar um alarme para três IPs. Entre eles, um IP carregava quase 2 GB de dados, o que é notável considerando que a média de bytes de upload está em torno de 500 MB. Porém, é importante observar que há casos em que IPs carregam 1 GB ou mais, portanto, disparar um alarme para esse IP está dentro da normalidade. Por outro lado, a regra bloqueou IPs que carregavam mais de 6 GB de dados, o que é uma quantidade substancial e merece ação imediata.

5.5 Novos Protocolos

Para aprimorar as capacidades de monitoramento, implementamos uma regra projetada especificamente para identificar o uso de novos protocolos que não foram observados anteriormente. A presença de novos protocolos pode indicar novos tipos de comunicação ou modificações no sistema.

Esta regra começa examinando os protocolos usados no conjunto de dados normal. Ele organiza os dados com base no tipo de protocolo (proto), calcula a contagem de cada protocolo e determina a representação percentual de cada protocolo no conjunto de dados.

Posteriormente, a regra analisa o conjunto de dados de teste para detectar novos protocolos. Seguindo um procedimento semelhante ao anterior, ele agrupa os dados por tipo de protocolo, calcula a contagem e a porcentagem para cada protocolo e mescla essas informações com os dados do protocolo do conjunto de dados normal. A regra retém apenas os registros onde a contagem de protocolo é zero no conjunto de dados normal, mas tem uma contagem positiva no conjunto de dados de teste.

```
1 # Verifique os protocolos utilizados
2 protocolos_normais = dados.groupby(['proto']).size().reset_index(nome='conta')
3 protocolos_normais['%'] = protocolos_normais['conta']/protocolos_normais['conta'].sum()
4 print("Protocolos normais: \n" + protocolos_normais.to_string(index=False))
5
6 # Verificar se há um novo protocolo em teste
7 protocolos_de_teste = teste.groupby(['proto']).size().reset_index(nome='conta')
8 protocolos_de_teste['%'] = protocolos_de_teste['conta']/protocolos_de_teste['conta'].sum()
9 protocolos_de_teste = pd.merge(protocolos_normais, protocolos_de_teste, on='proto', how='left')
10 protocolos_de_teste = protocolos_de_teste.fillna(0)
11 protocolos_de_teste = protocolos_de_teste[(protocolos_de_teste['conta_x'] == 0) & (protocolos_de_teste['%'] > 0)]
12 protocolos_de_teste = protocolos_de_teste[['proto', 'conta_você', '%']]
13 protocolos_de_teste = protocolos_de_teste.rename(columns={'conta_você': 'conta', '%': '% você'})
14 print("[ALARME] Novos protocolos em teste: \n" + protocolos_de_teste.to_string(index=False))
```

5.5.1 Teste de regras e identificação de dispositivos

Ao testar a regra, obtivemos os seguintes resultados:

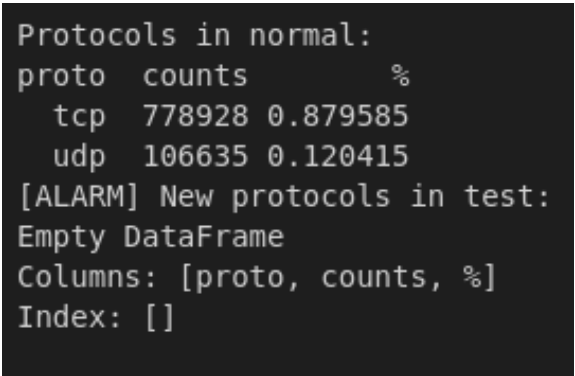


Figura 5.5: Resultado da regra de novos protocolos.

Como podemos ver, nenhum resultado foi encontrado porque nos conjuntos de dados fornecidos não foram utilizados novos protocolos. Contudo, seria importante implementar esta regra para garantir que a rede mantém o seu fluxo normal.

5.6 Novas Portas

Assim como na regra anterior (Novos Protocolos), seguimos a mesma metodologia para as portas utilizadas. O trecho de código detecta uma possível varredura de portas analisando os dados de tráfego da rede e comparando as portas usadas em condições normais com aquelas em um conjunto de dados de teste.

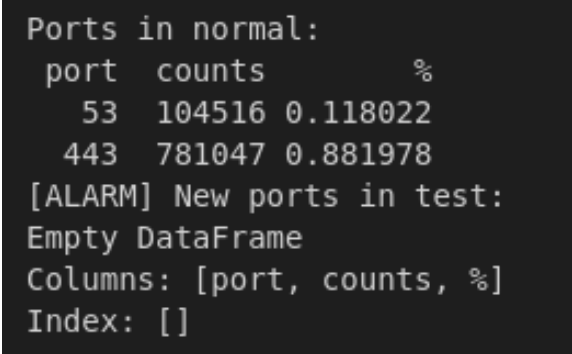
Na primeira etapa, o código categoriza os dados de tráfego de base por portas, contabilizando as ocorrências e calculando a participação percentual de cada porta. Os dados de teste passam pelo mesmo processo. O código então mescla os dois conjuntos de dados e filtra as portas que estão presentes apenas nos dados de teste. Estas novas portas, provavelmente não utilizadas em circunstâncias normais, são sinalizadas com uma etiqueta “ALARME” e apresentadas juntamente com as suas contagens e percentagens. A presença de novas portas pode indicar uma tentativa de varredura de portas ou outras atividades incomuns de rede, o que justifica uma investigação mais aprofundada.

```
1 # Verifique as portas usadas
2 portas_normais = data.groupby(['porta']).size().reset_index(nome='conta')
3 portas_normais['%'] = portas_normais['conta']/portas_normais['conta'].sum()
4 print("Portas normais: \n" + portas_normais.to_string(index=False))
5
6 # Verificar se há uma nova porta em teste
7 portas_de_teste = teste.groupby(['porta']).size().reset_index(nome='conta')
8 portas_de_teste['%'] = portas_de_teste['conta']/portas_de_teste['conta'].sum()
9 portas_de_teste = pd.merge(portas_normais, portas_de_teste, on='porta', how='left')
```

```
10 portas de teste = portas de teste.fillna(0)
11 portas de teste = portas de teste[(portas de teste['conta x'] == 0) &...
    (portas de teste['conta você'] > 0)]
12 portas de teste = portas de teste[['porta', 'conta você', '% você']]
13 portas de teste = portas de teste.rename(colunas={'conta você': 'conta', '% você':...
    '%'})
14 print("[ALARME] Novas portas em teste: ln" +...
    testar.portas.to_string(index=False))
```

5.6.1 Teste de regras e identificação de dispositivos

Ao testar a regra, obtivemos os seguintes resultados:



```
Ports in normal:
port  counts      %
53    104516 0.118022
443   781047 0.881978
[ALARM] New ports in test:
Empty DataFrame
Columns: [port, counts, %]
Index: []
```

Figura 5.6: Resultado da Nova Regra de Portas.

Como podemos ver, não há nenhum vestígio de varredura de portas.

Capítulo 6

Conclusão

O projeto demonstrou a importância crítica de regras robustas de SIEM no monitoramento de uma rede e na identificação de ameaças potenciais. Ao analisar o tráfego de rede típico de um dia, obtivemos informações valiosas sobre o uso normal da rede e as usamos para identificar desvios que poderiam indicar possíveis ameaças.

Ao testar essas regras SIEM em relação ao conjunto de dados “testX.parquet”, descobrimos que nossas regras foram eficazes para destacar comportamentos anômalos de rede, confirmando o valor de uma abordagem baseada em dados. Apesar da complexidade da tarefa, a utilização de pandas para análise de dados, a utilização de bases de dados para geolocalização baseadas em endereços IPv4 e a utilização de Jupyter Notebooks foram fundamentais para uma análise simples dos dados o que simplificou bastante a criação do SIEM regras.

O trabalho futuro deverá ter como objetivo refinar as regras com base na análise contínua da rede. Além disso, a integração de algoritmos de aprendizado de máquina poderia potencialmente aumentar a eficiência e a eficácia da detecção de comportamentos anômalos.