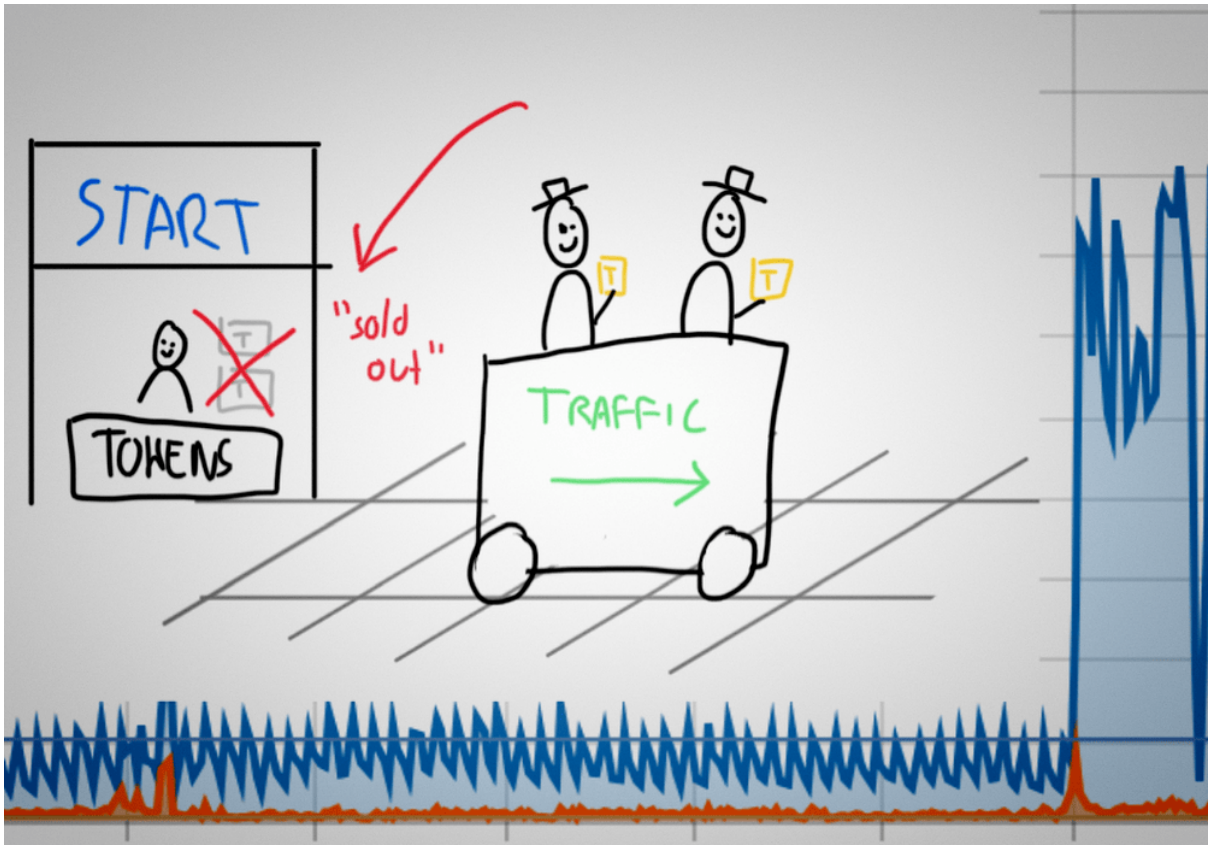
Verkeer vormgeven met iptables en tc

Beperken van uitgaande netwerkbandbreedte per client IP-adres

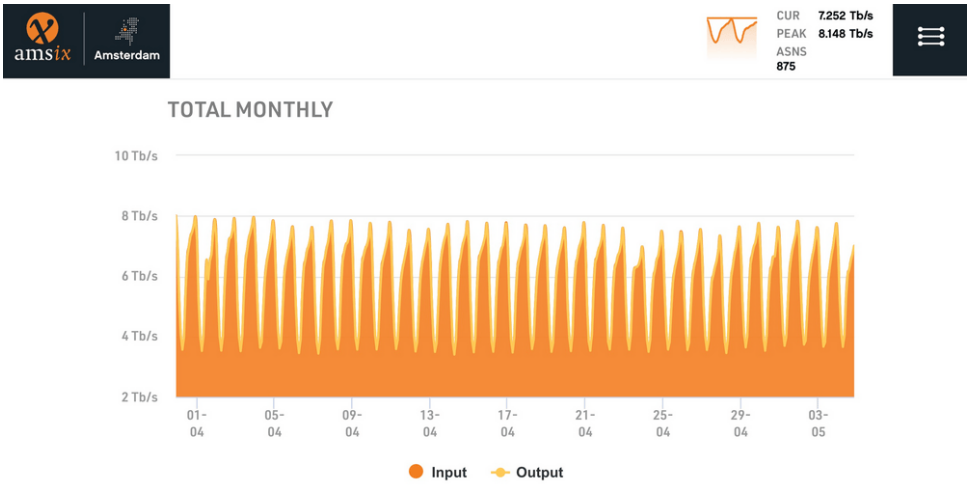
Willem L. Middelkoop
1 apr. 2020



Vorige maand ontving ik een geautomatiseerde melding over buitensporig bandbreedtegebruik, meestal een teken van problemen. Wanneer dit gebeurt, dien je een standaard incidentenprocedure te volgen, waarbij je probeert de bron van het verkeer te isoleren voordat je het afsluit. De oorzaak van dit incident was echter niet wat ik verwachtte... waardoor een ander soort oplossing nodig was dan een simpele blokkade.

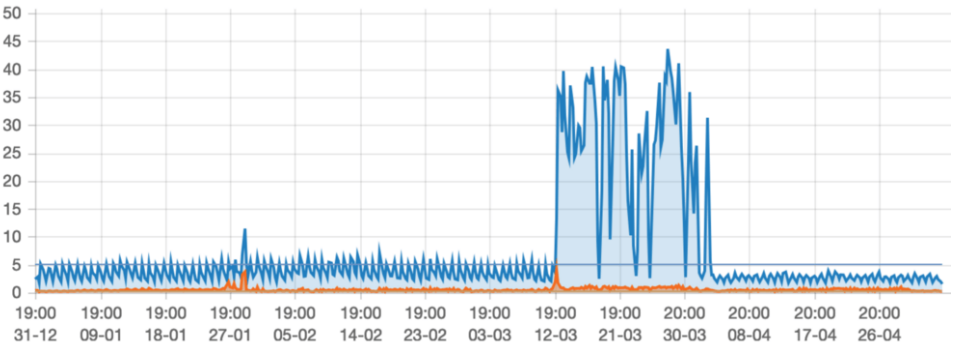
Excessief bandbreedte alarm

Wanneer je servers beheert zul je merken dat internetverkeer meestal in voorspelbare patronen plaatsvindt. Net zoals echt verkeer op de wegen, zijn er regelmatige tijden waarop het druk en rustig is.



Bandbreedtegraafiek van AMS-IX toont een voorspelbaar patroon - merk het golfachtige patroon op

Het voorspelbare patroon maakt het mogelijk om anomalieën automatisch te detecteren. Net zoals een file op de snelweg op een ongebruikelijk tijdstip het gevolg kan zijn van een ongeluk, kunnen onverwachte schommelingen in internetverkeer een indicator zijn voor een incident in de cyberruimte.



Bandbreedtegraafiek met ongebruikelijke piek die aangeeft dat er iets mis is - je hoeft geen Sherlock Holmes te zijn om het te vinden

De bron isoleren

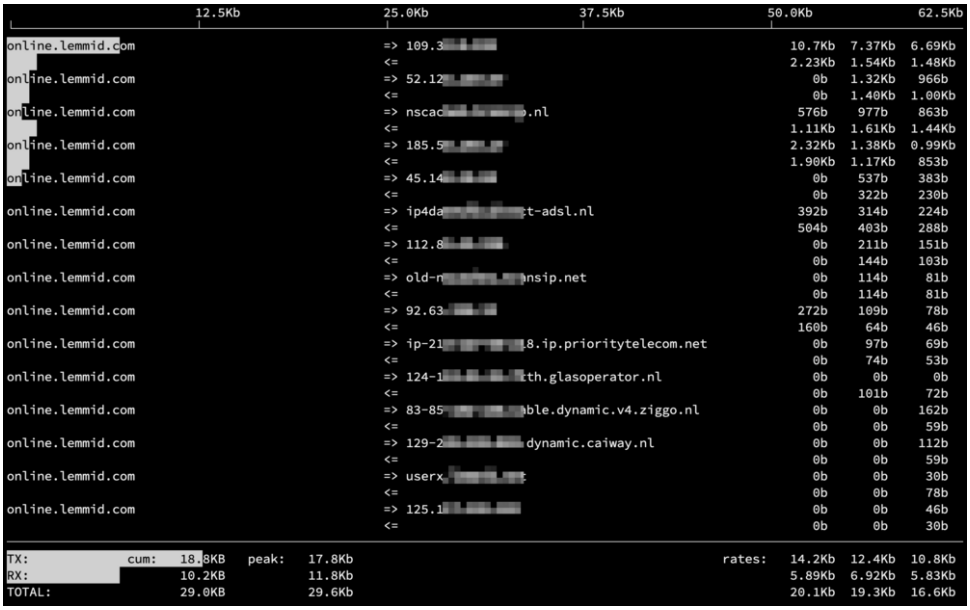
Het eerste wat je moet doen is de bron van het probleem vinden. Doe dit door de anomalie te analyseren door:

- Vast te stellen of het extra verkeer inkomend of uitgaand is (upload of download?)
- De bijbehorende bron- en doel-IP-adressen te bepalen (welke server wordt getroffen? Waar gaat het verkeer naartoe of waar komt het vandaan?)
- Het soort verkeer te bepalen (e-mail, web of iets anders?)

Om dit te doen, moet je verkeersgrafieken inspecteren, deze geven meestal een duidelijk onderscheid tussen input/output aan. Vervolgens moet je proberen de getroffen IP-adressen te vinden. Dit kan inhouden dat je meer (individuele server) grafieken en statistieken in switches, routers en servers bekijkt. Kijk vervolgens naar CPU-gebruik

en logbestanden om te bepalen welke applicatie wordt getroffen, zoals mail of web. Hoe groter de anomalie, hoe gemakkelijker deze te vinden is.

Je kunt in de verleiding komen om de anomalie te stoppen zodra je hem hebt gevonden, door het bijbehorende verkeer onmiddellijk te stoppen. Maar je moet het echt in leven houden (in ieder geval nog even) om er zoveel mogelijk van te leren.



iftop tool gebruiken om bandbreedtegebruik per verbonden IP-adres te bekijken

Gebruik een tool zoals iftop om een realtime overzicht te zien van het bandbreedtegebruik per verbonden IP-adres. Het is erg handig om een idee te krijgen van wat er gaande is, vooral in combinatie met logbestanden (het koppelen van IP-adres aan individuele accounts).

Ongebruikelijke oorzaak

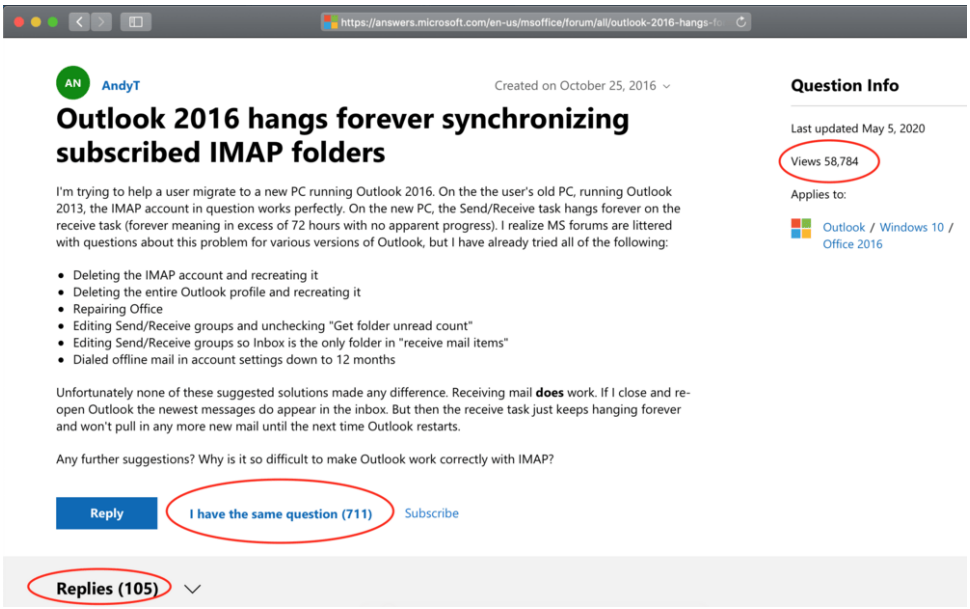
De server die het excessieve bandbreedtegebruik produceerde was een mailserver. Vaak worden ze door hackers gebruikt om ze in een spam relay server te veranderen. Dit kan op verschillende manieren gebeuren, meestal door het verkrijgen van een geldige login door de spammers. Dit resulteert in veel uitgaand verkeer omdat de spammers veel e-mails zullen versturen. Als er een spam-run op je server plaatsvindt, zie je dit waarschijnlijk in de logs terug doordat spamberichten vaak bouncen, waardoor de mailqueues snel vollopen. Het wordt snel een rommeltje, maar op deze server was er geen sprake van zo'n rommeltje - alleen veel ongebruikelijk bandbreedtegebruik.

Mail (en spam) wordt verzonden met behulp van het SMTP-protocol, maar het verkeer op de SMTP-poort van deze server was heel normaal. Dit was zeer ongebruikelijk, omdat het betekende dat het excessieve verkeer ergens anders vandaan kwam. Maar wat? Na het analyseren van verschillende logbestanden op de mailserver heb ik vastgesteld dat het betreffende protocol IMAP was. Op de een of andere manier veroorzaakte een client die via IMAP met de mailserver verbonden was, overmatig veel netwerkverkeer. IMAP is een protocol dat wordt gebruikt om e-mail te *lezen*, niet om het te verzenden, waardoor IMAP-gerelateerde anomalieën zeer zeldzaam zijn.

Door de log van de mailserver te analyseren, vond ik de specifieke gebruiker die het verkeer veroorzaakte. Ik nam contact op met de klant om te vragen of hij iets vreemds had gemerkt. Niet verrassend, hij merkte dat zijn computer de laatste tijd een beetje traag aanvoelde.

Microsoft Outlook synchronisatielus

Na wat proberen via de telefoon, kwamen we erachter dat iets ervoor zorgde dat Microsoft Outlook IMAP-mappen in een lus bleef synchroniseren. Dit zorgde ervoor dat zijn computer de volledige inhoud van zijn mailbox steeds opnieuw downloadde! Omdat zijn mailbox meer dan 40 gigabyte groot was, veroorzaakte dit het aanzienlijke verkeer. Blijkbaar wordt dit veroorzaakt door een bug in Microsoft Outlook, helaas is er geen gemakkelijke oplossing voor.



Bugs in Microsoft Outlook zorgen ervoor dat het IMAP-mappen blijft synchroniseren, een probleem dat veel mensen ervaren (zie het aantal weergaven!)

Beperken door traffic shaping

Ik stond voor de moeilijke beslissing om de (normale, legitieme, betalende) klant te blokkeren of om het excessieve verkeer te laten doorgaan (met aanzienlijke kosten voor de netwerkprovider). Het blokkeren van legitieme gebruikers die hun bedrijf via e-mail runnen is een zeer slecht idee, maar het toestaan van excessief verkeer is ook slecht. Gelukkig vond ik een alternatieve manier om het excessieve verkeer te verminderen terwijl de klant toch toegang tot zijn mail behield (met behulp van zijn vertrouwde, maar gebrekkige Outlook-app).

Traffic Shaping

Als bandbreedtebeheertechniek kun je traffic shaping gebruiken om sommige (of alle) datapakketten te vertragen om ze in overeenstemming te brengen met een gewenst verkeersprofiel. Indien toegepast, vorm je - letterlijk - de verkeersgrafieken, vandaar de naam.

Deze techniek is niet zonder controverse, omdat het nogal het tegenovergestelde is van het vaak geprezen "net neutraliteit" principe. Met traffic shaping kun je opzettelijk specifieke soorten verkeer blokkeren of vertragen (of extra geld in rekening brengen). In principe ben ik tegen alles wat de netneutraliteit schaadt, maar voor deze specifieke situatie had ik geen haalbaar alternatief. Ik moest het e-mailverkeer voor deze specifieke klant ernstig vertragen, waardoor de hoeveelheid bandbreedte werd verminderd, terwijl hij toch toegang tot zijn account behield.

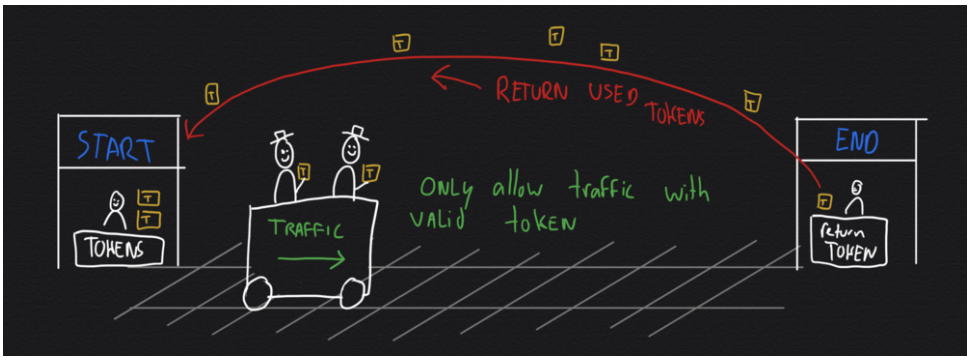
Implementatie van traffic shaping

Om verkeer te vormen, moet je twee dingen doen:

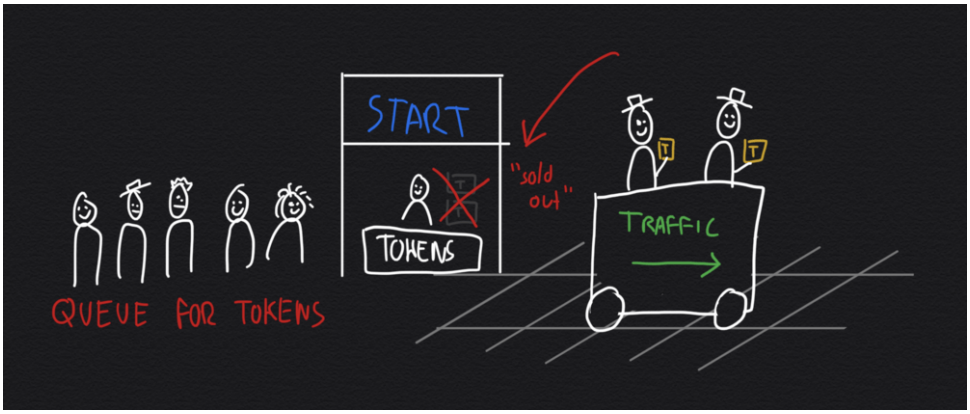
- **verkeer markeren**: Je wilt alleen bepaald netwerkverkeer beïnvloeden, in dit geval IMAP-toegang voor een bepaalde client. Ander verkeer mag niet worden beïnvloed. Dit wordt gedaan door pakketten te markeren die voldoen aan bepaalde kenmerken, zoals client-IP of poortnummers
- **vormbeleid afdwingen**: Met behulp van verkeerscontroletools zoals 'tc' dwing je vervolgens het vormbeleid af op het gemarkeerde verkeer. Er zijn verschillende manieren om dit te doen, maar een veelgebruikte manier is om te werken met zogenaamde "Hierarchy Token Buckets" of HTB's.

Hierarchy Token Buckets (HTB's)

In principe is een token bucket vergelijkbaar met het principe van het beperken van het aantal passagiers in een treinrit door slechts een vast aantal beschikbare tokens uit te delen. Wanneer passagiers (of datapakketten) de trein (of het netwerk) binnenkomen, nemen ze een token. Wanneer ze uitstappen (of hun bestemming bereiken), wordt de token teruggegeven. Met behulp van het token bucket principe kun je de hoeveelheid gelijktijdig verkeer in een systeem regelen.



Tokens gebruiken om verkeer te controleren - alleen passagiers (of datapakketten) met een geldige token zijn toegestaan. Tokens worden teruggegeven als het verkeer zijn bestemming bereikt.



Verkeer moet wachten tot tokens beschikbaar komen wanneer het maximale aantal tokens is weggegeven, waardoor het maximale gelijktijdige verkeer wordt afgedwongen

Je kunt dit principe implementeren met behulp van de netwerktools ”tc” (voor verkeerscontrole) in combinatie met ”iptables”. Je kunt een script gebruiken om de regels voor markeren en afdwingen in te stellen. Je kunt verschillende voorbeelden online vinden, ik heb deze gebruikt door Julien Vehent.

```
#! /bin/bash
NETCARD=eth0
MAXBANDWIDTH=100000  # choose a number that is high enough for non-shaped traffic

# reinit
tc qdisc del dev $NETCARD root handle 1
tc qdisc add dev $NETCARD root handle 1: htb default 9999

# create the default class, this is "all the other traffic"
tc class add dev $NETCARD parent 1:0 classid 1:9999 htb rate $(( $MAXBANDWIDTH ))kbit ceil $(( $MAXBANDWIDTH ))kbit burst 5k prio 9999

# control bandwidth per IP
declare -A ipctrl
# define list of IP and bandwidth (in kilo bits per seconds) below
ipctrl[192.168.1.101]="128"             # limited to 128 kilobits per second
ipctrl[192.168.1.102]="512"             # limited to 512 kilobits per second
ipctrl[192.168.1.103]="32"              # limited to just 32 kilobit per second

mark=0
for ip in "${!ipctrl[@]}"
do
    mark=$(( mark + 1 ))
    bandwidth=${ipctrl[$ip]}

    # traffic shaping rule
    tc class add dev $NETCARD parent 1:0 classid 1:$mark htb rate $(( $bandwidth ))kbit ceil $(( $bandwidth ))kbit burst 5k prio $mark

    # netfilter packet marking rule
    iptables -t mangle -A INPUT -i $NETCARD -s $ip -j CONNMARK --set-mark $mark

    # filter that bind the two
    tc filter add dev $NETCARD parent 1:0 protocol ip prio $mark handle $mark fw flowid 1:$mark

    echo "IP $ip is attached to mark $mark and limited to $bandwidth kbps"
done

#propagate netfilter marks on connections
iptables -t mangle -A POSTROUTING -j CONNMARK --restore-mark
```

Voorbeeld traffic shaping script

Conclusie

Soms zijn netwerkanomalieën niet wat je ervan verwacht, daarom moet je altijd de tijd nemen om ze te onderzoeken! Het blindelings blokkeren van verkeer is bot, soms moet je je methoden verfijnen om problemen te beperken.

Het toepassen van onorthodoxe filtertechnieken is niet iets dat ik leuk vind, maar soms zijn ze het enige middel tot een doel. Weten wat je doet en waarom je het doet is erg belangrijk!