heinlein

Willkommen

Linux höchstpersönlich.

heinlein

12.03.2023 / Carsten Rosenberg, Manu Zurmühl

Enterprise Grade Mailcluster mit Open-Source?

heinlein

Wer wir sind

Carsten Rosenberg
Linux-Consultant

Manu Zurmühl
Linux Consultant

E-Mail und seine Probleme

- Privat werden heute vor allem Messenger Dienste genutzt
- E-Mail ist aber das neue Brief und Fax
- E-Mail Haupteinfallstor für Spam, Phishing und Malware
- Durch E-Mails wurden zahlreiche Sicherheitsvorfälle
- Fälschbarkeit Absender und Inhalts einer Mail ist problematisch
- SMTP Standard aus 1982
- Kein überarbeitetes Protokol - neue Entwicklungen als Erweiterung
- Erweiterungen sind wie immer optional
- Wildwuchs an Implementierungen
- Viele wägen bessere Sicherheitsmechanismen gegen die Angst verlorener Mails ab

SMTP RFC's

- RFC 821 (SIMPLE MAIL TRANSFER PROTOCOL)
- RFC 2821 (SIMPLE MAIL TRANSFER PROTOCOL)
- RFC 1845 (SMTP Service Extension for Checkpoint/Restart)
- RFC 1870 (SMTP Service Extension for Message Size Declaration)
- RFC 1869 (SMTP Service Extensions)
- RFC 1894 (An Extensible Message Format for Delivery Status Notifications)
- RFC 1985 (SMTP Service Extension for Remote Message Queue Starting – ETRN)
- RFC 2034 (SMTP Service Extension for Returning Enhanced Error Codes)
- RFC 2047 (Message Header Extensions for Non-ASCII Text)
- RFC 2487 (SMTP Service Extension for Secure SMTP over TLS)
- RFC 2505 (Anti-Spam Recommendations for SMTP MTAs)
- RFC 2554 (SMTP Service Extension for Authentication)
- RFC 2606 (Reserved Top Level DNS Names)
- RFC 2852 (Deliver By SMTP Service Extension)
- RFC 2920 (SMTP Service Extension for Command Pipelining)
- RFC 3030 (SMTP Service Extensions for Transmission of Large and Binary MIME Messages)
- RFC 3207 (SMTP Service Extension for Secure SMTP over Transport Layer Security)
- RFC 3461 (SMTP Service Extension for Delivery Status Notifications (DSNs))
- RFC 3463 (Enhanced Status Codes for SMTP)
- RFC 3464 (An Extensible Message Format for Delivery Status Notifications)
- RFC 3700 (Internet Official Protocol Standards)
- RFC 3974 (SMTP Operational Experience in Mixed IPv4/v6 Environments)
- RFC 4409 (Message Submission for Mail, führt Port 587 für Message Submission ein)
- RFC 5321 (Simple Mail Transfer Protocol)
- RFC 5322 (Internet Message Format)
- RFC 5336 (SMTP Extension for Internationalized Email Addresses)
- RFC 6409 (Message Submission for Mail, Internet Standard, löst RFC 4409 ab)
- RFC 6152 (SMTP Service Extension for 8bit-MIMEtransport)
- RFC 7505 (A "Null MX" No Service Resource Record for Domains That Accept No Mail)
- RFC 8314 (Cleartext Considered Obsolete: Use of Transport Layer Security (TLS) for Email Submission and Access)

SMTP

- === Trying rspamd.de:25...
- === Connected to rspamd.de.
- <- 220 mail.rspamd.de ESMTP Postfix
- -> HELO rspamd.de
- <- 250 mail.rspamd.de
- -> MAIL FROM:<root@rspamd.de>
- <- 250 2.1.0 Ok
- -> RCPT TO:<root@rspamd.de>
- <- 250 2.1.5 Ok
- -> DATA
- <- 354 End data with <CR><LF>.<CR><LF>
-
- -> Date: Wed, 08 Mar 2023 13:43:44 +0100
- -> To: root@rspamd.de
- -> From: root@rspamd.de
- -> Subject: test Wed, 08 Mar 2023 13:43:44 +0100
- -> Message-Id: <20230308134344.3402816@rspamd.de>

Was war damals wichtig:

- Netzwerkbandbreite sparen
- Hauptsache eine Mail kommt an
- Vertrauensbeziehung zwischen den Teilnehmern
- Gefälschte Daten nicht relevant

Spam Entwicklung

- Spam was introduced by Hormel on July 5, 1937.[6]
- The Oxford Encyclopedia of Food and Drink in America states that the product was intended to increase the sale of pork shoulder which was not a very popular cut.[6]
- (https://en.wikipedia.org/wiki/Spam_(food))

Spam Entwicklung

- 1864 Complaints about unwanted Telegraph messages in the Times newspaper
  - https://blog.knowbe4.com/here-is-a-spam-message-from-1864-as-old-as-the-victorian-internet

  In the late 19th Century Western Union allowed telegraphic messages on its network to be sent to multiple destinations. The first recorded instance of a mass unsolicited commercial telegram is from May 1864, when some British politicians received an unsolicited telegram advertising a dentist.
- 1937 introduction of Spam (Meat)

  Spam was introduced by Hormel on July 5, 1937.[6] The Oxford Encyclopedia of Food and Drink in America states that the product was intended to increase the sale of pork shoulder which was not a very popular cut.[6] (https://en.wikipedia.org/wiki/Spam_(food))
- 1978 first unwanted newsletter from a DEC marketer to dozens of people in the ARPANET
- 198x's usage of the term spamming in Multi User Dungeons – text based
  https://en.wikipedia.org/wiki/Spam_(Monty_Python)
- 1993 by accident 200 mails to a USENET group -> first time called spam
- 1994 commercial spam to USENET "Green Card Lottery- Final One?"
- 1997 Blocking Spam with MAPS "blackhole list"

Spam Entwicklung

- 1998 First DNS based RBL's
- 1999 "Happy99" worm, "Melissa" worm
- 2000 "Iloveyou" Outlook Worm
- 2001 "Anna Kournikova virus" Outlook Worm
- 2002 Paul Graham "A plan for spam" Bayesian filtering
- 2004 Bill Gates announced that "spam will soon be a thing of the past."
- 2004 first postgrey release
- 2005 Idea of phishing using ebay.com fake mails
- 2006 IronPort released a study which found 80% of spam emails originating from zombie computers.
- 2008+ Spam got more dynamically
  - Daily changing campaigns
  - Targeted phishing waves
  - But still - viagra spam

<https://en.wikipedia.org/wiki/Spamming>

How to avoid Spam in the 90's

- Add some manual rules to your MTA
- Add an IP Block List

How to avoid Spam in 2000+

- Add many rules manual rules to your MTA
  https://www.postfixbuch.de/upload/header_checks
- Add multiple RBL's
- Add a mail content filter like spamassassin
- Add an Antivirus scanner
- Add Greylisting
- Write additional rules for spamassassin
- Train mails in Spamassassins Bayes filter

How to avoid Spam in 2010+

- Puh, let’s say its getting more complicated …

Spam bis Ende der 2000er

- unpersonalisierte Massenmails - "Viagra" / "Green Card" Spam
- gehackte Accounts und Server sowie (Windows-) Bots
- Viren über lange Zeit gleich

Spam heute

- Bulk Spam + targeted Spam
- Grundlast typischer Massen-Spam
- Schnell ändernde Spam Templates
  - leicht andere Formulierung
  - Anhängen von irrelevantem Text (z.B aus Wikipedia)
  - Subject - einfach mal ein Re: anhängen
- Antwort-Mails auf alte Kommunikation
  - Geknackte Postfächer, Exchange ProxyShell
  - Mails sind recht fehlerfrei und haben Relevanz zur alten Kommunikation

Spam heute

- Targeted Spam komplexer
  - CEO-Spam ist auch mal handgeschrieben
  - Unterscheidung zwischen verified und unverified Adressen
  - sehr gute Phishing Kampagnen
- Spamkampagnen werden auch mit Anti-Spam Tools verfiziert
- Geknackte Accounts werden vorsichtig angetestet
- Eingehende und ausgehende Spam Kampagnen werden bei großen Mail-Providern vorab getestet
- personalisierte Anhänge, on-the-fly generierte Viren
- Spammer adaptieren E-Mail Schutzmechanismen
- verschlüsselte Anhänge

Ja auch eine Folie mit ChatGPT

- KI Chat-Bots können natürlich auch Spam/Phishing Vorlagen erstellen
- Antworten auf abgezogene E-Mails könnten noch besser formuliert werden
- Chat-Bot könnte auch die Kommunikation mit dem Opfer übernehmen
- CEO-Spam / BEC mit Branchen-News
- Bing-Bot gehijackt
  https://www.zeit.de/digital/2023-03/cyberangriffe-microsoft-bing-chat-piraten
- Facebooks LLaMa samt Parameter geleaked

Naja, zum Glück ist E-Mail ja bald tot

- Hauptkommunikationsform von Firmen
- E-Mail Server werden ständig angegriffen, sind aber meist weniger gut geschützt
- E-Mail Security sollte heutzutage wie alle anderen Bereiche im IT-Sicherheitskonzept vorgesehen sein und auch aktiv ausgewertet werden
  - Es sind nicht mehr nur nervige Viagra Mails
  - Gutes Phishing erkennt auch keine fancy Firewall
- E-Mail und Websicherheit sollten Hand in Hand zusammen arbeiten

Wie weiter?

- Für die Absicherung einer Mail-Infrastruktur gibt es schon ein paar gute Konzepte
- Was sagen denn die kommerziellen Anbieter was wir genau brauchen?

Buzzword Bingo (alles nur geklaut)

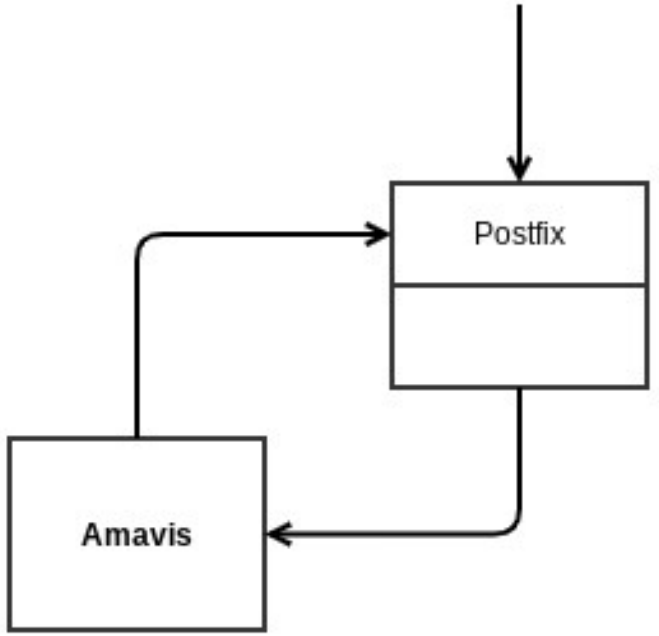
- Adaptive Ratelimiting
- Advanced Anomaly Detection
- Advanced Malware Protection and Threat Grid
- Advanced Multi-layer Attack Vector Detection
- Advanced Multi-layer Malware Detection
- Advanced Threat Protection
- Artificial Intelligence (AI) Spam Detection
- BEC and CEO Fraud Detection
- Cloud-driven Reputation
- Content Disarm and Reconstruction
- Domain Fraud Protection

Buzzword Bingo (alles nur geklaut)

- Email Data Loss Prevention
- Identity-Based Encryption (IBE)
- Impersonation Analysis
- Local and Cloud Sandboxing
- Mailbox Safeguard
- Recipient dependend Transport Layer Security and Encryption
- SIEM Vector Export
- Typosquatting Detection
- URL Click Protection
- Web Interaction Tracking
- Secure Message Delivery
- Virus Outbreak Protection

Mail Infrastruktur früher

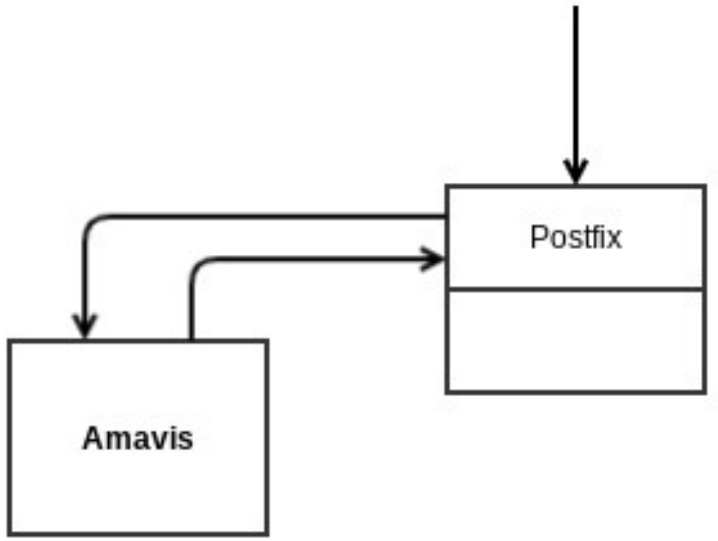
- Antispam / Antivirus ist ressourcenintensiv daher Post-Queue (nach Annahme der Mail)
- Daher lieber früh ablehnen
- Entscheidungen im MTA sind aber meist einzelne Abfragen, die direkt zu einer Aktion führen
  - IP ist auf der RBL
  - Subject "Viagra kaufen"
- Jeder MTA arbeitete für sich allein
- Nach der Annahme der Mail wurden Antispam / Antivirus Tools aufgerufen, die unabhängig vom SMTP Dialog geprüft haben



Mail Infrastruktur später

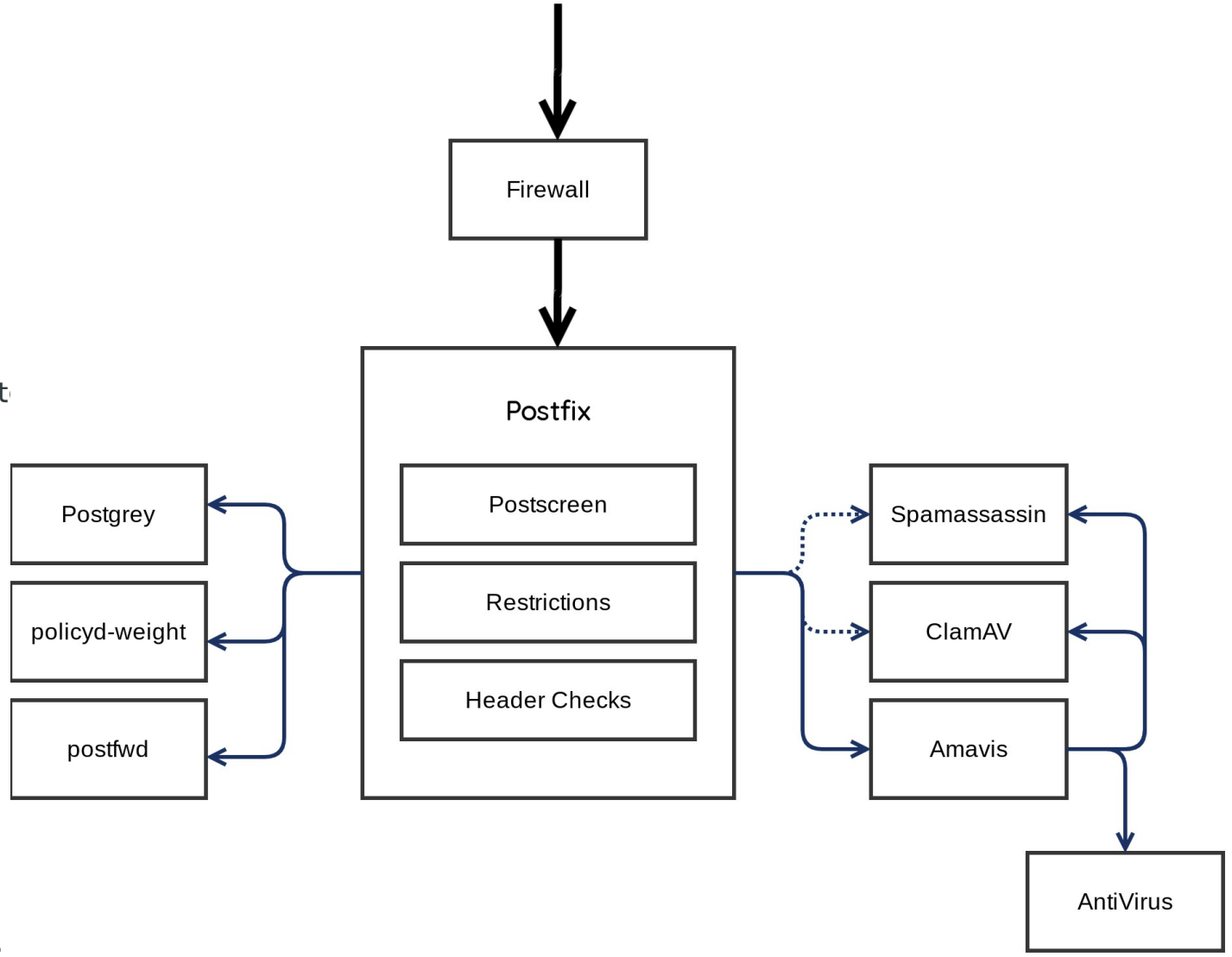
- Antispam / Antivirus parallel zur Postfix Verarbeitung ausgeführt

  (Pre-Queue via SMTP Proxy Filter bzw Milter)
- Ablehnung durch Postfix erfolgte trotzdem am Besten sehr früh, damit Spam nicht komplett gescannt werden muss



Anti-Spam mit Postfix

- Postfix Restrictions
  - lokale Access Maps
  - Mail RFC Regeln (reject_non_fqdn_sender)
  - RBL / RHSBL
- Postfix Header/Body Checks
  - /Subject: Viagra günstig kaufen/i REJECT
- Greylisting als Postfix Restriction Policy Daemon (Post
- Policyd-Weight / Postfwd / Postfix Policyd
  - gewichtete DNS / RBL Prüfung
  - Ratelimits
  - Block- / Allow-Lists etc.
- Postscreen
- Spamassassin / Clamav
- später Amavis (inkl. Spamassassin und AV)
- Einbindung kommerzieller Appliances



Und dann kam Rspamd ins Spiel

- seit 2018 bei uns
- initial als 1:1 Ersatz für Amavis (+ Spamassassin)
- Alle Indikatoren Postfix, Amavis und ClamAV jetzt in einem Tool
- Rspamd hat eine zentrale Sicht auf den gesamten Mailflow
- Spannend war es dann für uns aus den ganzen Indikatoren einen Vorteil ziehen zu können
- Prüfungen in Rspamd verlagert
- Dann haben wir gesehen, dass wir auch Routings, Adressumschreibungen oder Policies (wer darf was) im Rspamd viel leichter und flexibler umsetzen können

Und dann kam Rspamd ins Spiel

- Vorher aber Aufräumen:
  - Vom Postfix abgelehnte Mail sieht der Rspamd nicht
  - Nicht gesehene Mail kann nicht analysiert/gelernt werden
  - viel Potenzial bleibt liegen
- Postfix wurde um viele Sachen erleichtert, die der Rspamd auch kann:
  - Postscreen
  - RBL
  - Policyd-Weight
  - Postgrey (Greylisting)

Anti-Spam mit Rspamd und Postfix

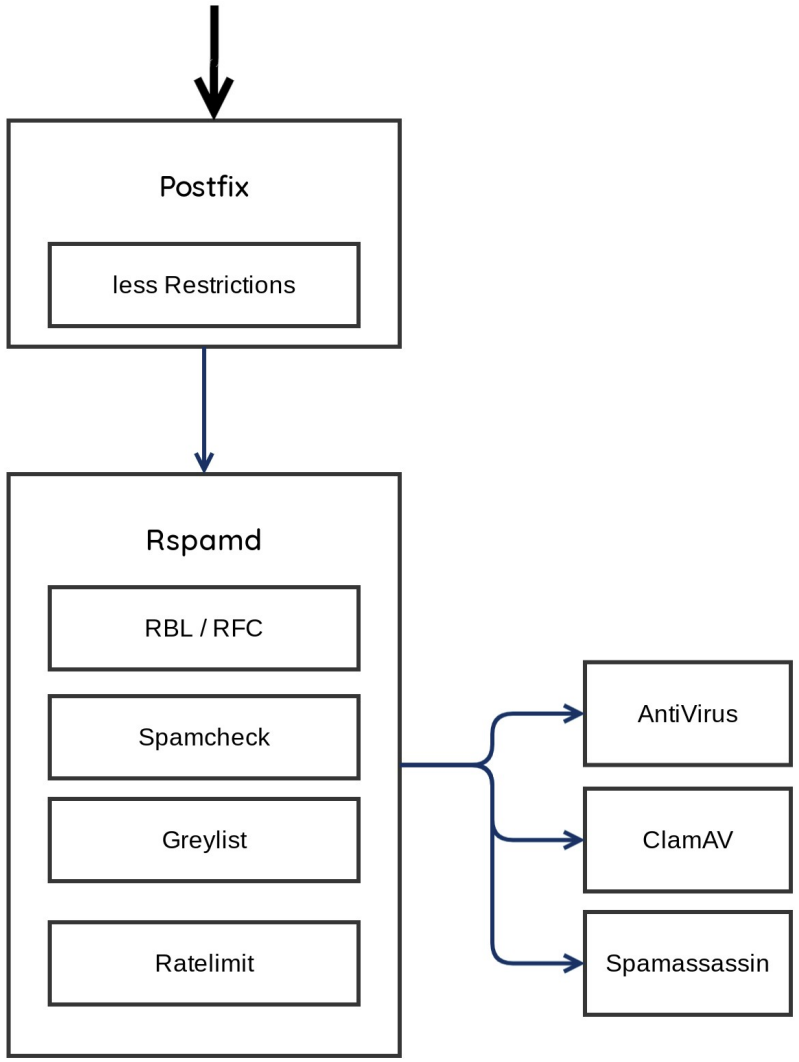
Postfix

- Grundlegendes Routing
  - erlaubte Domains und User
  - statisches Routing
  - statische Adressumschreibung (virtual)
    - Alias-Adressen der User sind in Dovecot besser aufgehoben

Anti-Spam mit Rspamd und Postfix

Rspamd

- Anti-Spam / Antivirus
  - Header / Body Checks
  - Mail RFC
  - DNS Prüfungen
- Block- / Allow-Lists
  - RBL / RHSBL (URIBL/SURBL)
- eigene Reputationen
- Greylisting
- Ratelimit
- Abfrage von Appliances oder Firewalls
- Einbindung von Sandboxing und SIEM Tools

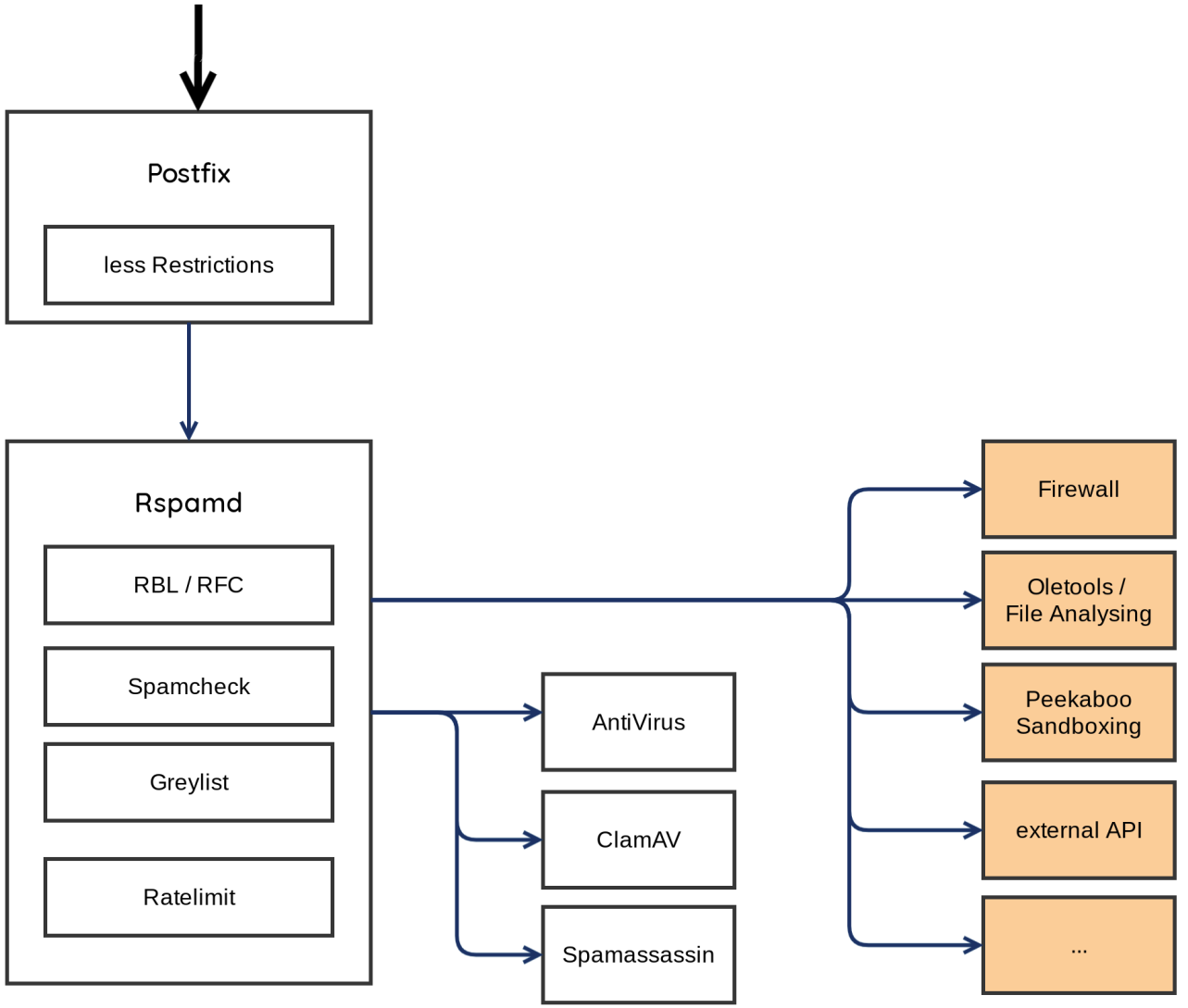


Anti-Spam mit Rspamd und Postfix

Rspamd

- API Abfragen - wie Dovecot Quota
- User Informationen via Datenbankabfrage
- eigene Maschine Learning Auswertungen
- dynamisches Routing
- dynamische Adressumschreibung
- Anwendungen lokaler Policies

- UND: wir machen alles **PRE-Queue**



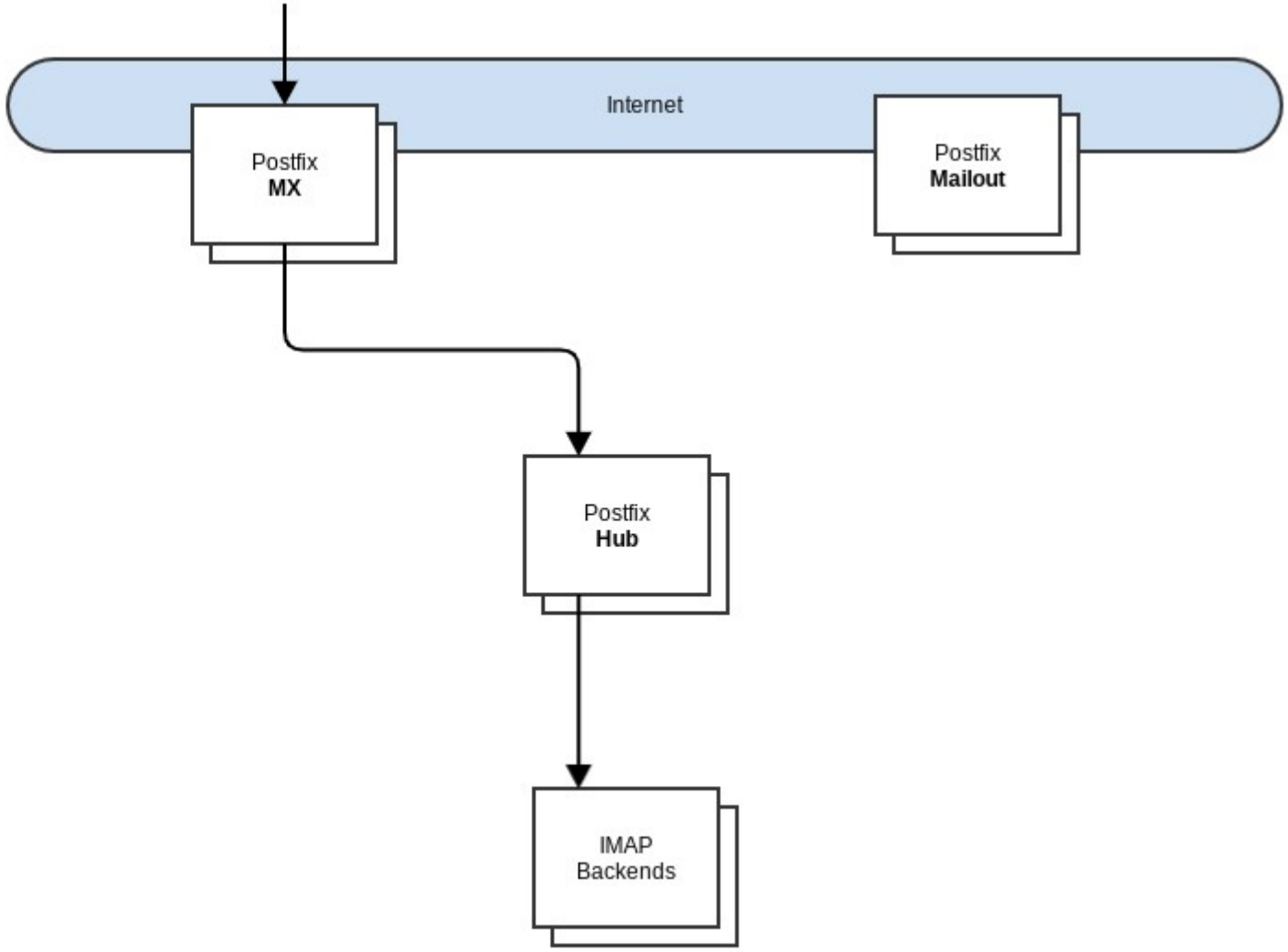
Rspamd AntiSpam in der neuen Mail Infrastruktur

- Postix bildet das Grundgerüst
  - einfache Setups
  - keinerlei komplexe Routings oder Umschreibungen
- Rspamd ist das Hirn und Herz
  - Verlagerung wichtiger Prüfungen in den Rspamd
  - Abfragen gegen jede erdenkliche externe Quelle
  - viel detailliertere Prüfungen
  - alle Informationen werden gewichtet und aggregiert
  - kein Einzelergebnis führt unausweichlich zu einem Reject
  - Profile / Policies steuern verschiedene Aktionen
  - Rspamd kennt den Zustand einer Mail an jedem Hop der Infrastruktur
  - Rspamd bewertet nicht nur
  - Adressumschreibungen
  - Routing einzelner Adressaten

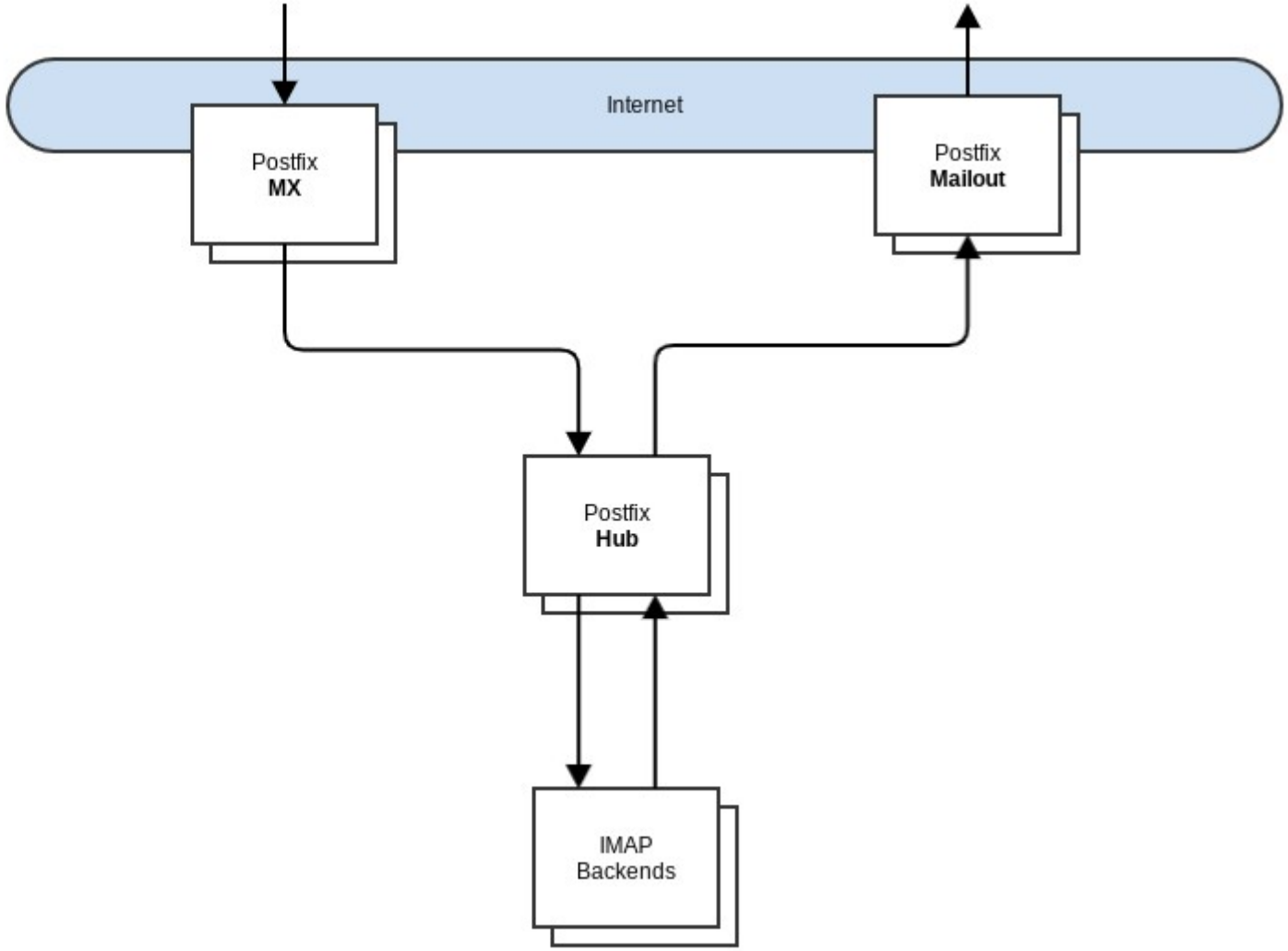
Kurz Vorab: unsere "Cluster-Zeichnung"

- Aufbau der Mail Cluster mit getrennten internen und externen Systemen
- jedes bei uns einzeln dargestellte System würde in größeren Umgebungen natürlich redundant ausgelegt sein
- Auch wenn wir hier Cluster mit 10-15 Systemen darstellen - alle genannten Mechanismen lassen sich auch auf einem Single- oder Dual-System umsetzen
- die Aufteilung einzelner Mechanismen erfolgt oft um eine sauberere Konfiguration zu haben oder Dienste aus (sicherheitstechnischen) Gründen zu trennen

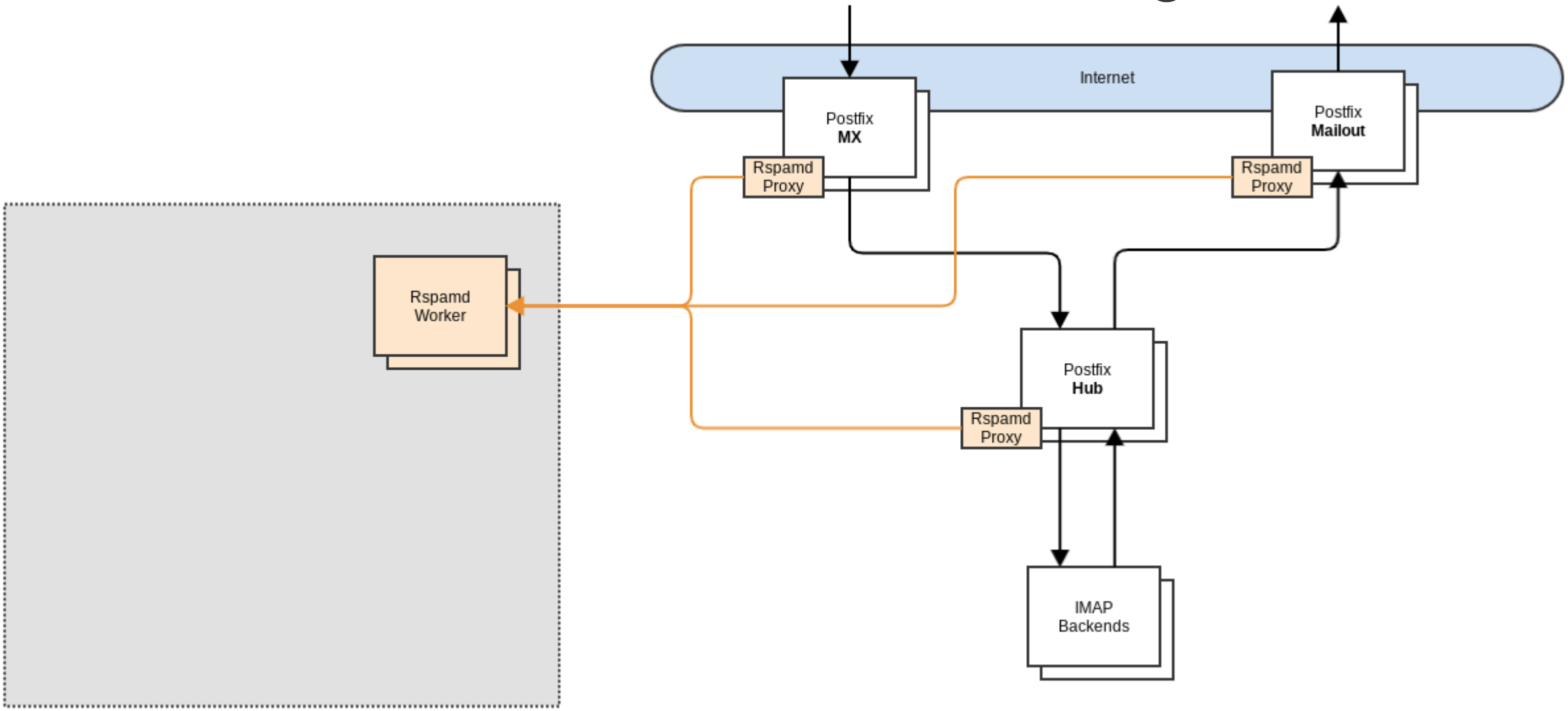
Kurz Vorab: unsere "Cluster-Zeichnung"



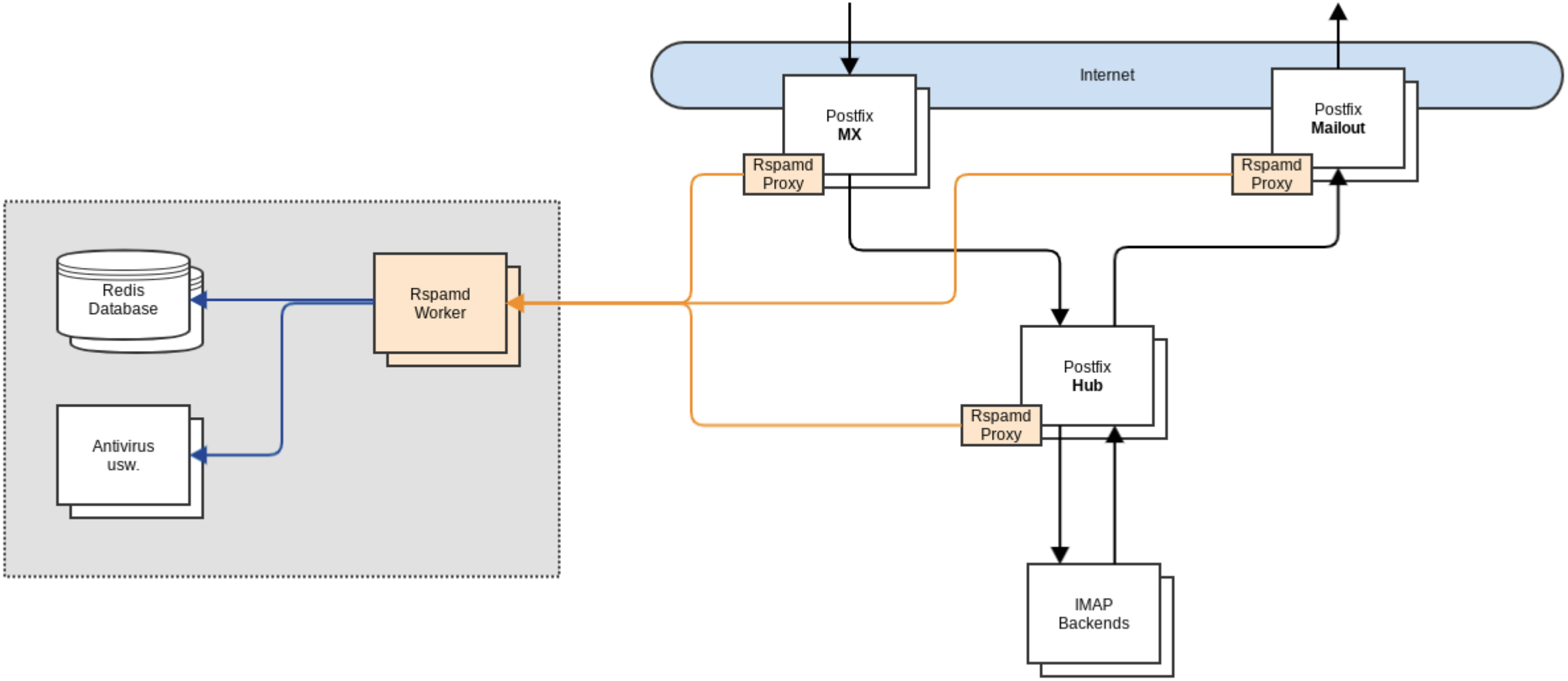
Kurz Vorab: unsere "Cluster-Zeichnung"



Kurz Vorab: unsere "Cluster-Zeichnung"



Kurz Vorab: unsere "Cluster-Zeichnung"



Ein paar Details vom Rspamd

- Seit ca. 2008 eher wirklich ab 2015 entwickeltes Anti-Spam Tool
- Vom Feature-Set her eher ein Mail-Processing Framework
- High Availability schon eingebaut
- setzt im Gegensatz zu Spamassassin weit weniger auf statische Regeln
  - lieber gut gemachte Funktionen
  - eigene Reputationen
  - besseres Lernen
  - Maschine Learning zur Historischen Auswertung

Was steckt im Rspamd selbst

- ganz am Anfang Rspamd konvertiert die Mail und bereitet die ganzen Informationen zur späteren Nutzung auf
- die relevanten Informationen stehen danach für Plugins und Funktionen bereit
  - task:is_html() -> Boolean, ist eine HTML Mail
  - task:get_urls() -> Gibt alle gefundenen URLs zurück
- Settings (-Profile) können nach sehr flexiblen Kriterien bestimmte Profile mit angepassten Einstellungen triggern
  - Scan ohne eine bestimmte RBL oder eines bestimmten Antivirus
  - Ein- / Ausgehend unterschiedliche Prüfungen durchführen
  - ...
- Abarbeitung des mitgelieferten Regelwerks
- Abfragen verschiedenster Werte gegen (externe) Listen / DBs
- Abfrage verschiedenster Werte gegen RBL's
- Abfrage von Antiviren Software und externen Analyse-Tools

Was steckt im Rspamd selbst

- Abfrage und Update verschiedenster Werte gegen eine eigene Reputationsdatenbank
- Text-Analyse mit Bayes und Fuzzy Algorithmen
- Erkennung ähnlicher Scans und deren Einordnung via neuronalem Netzwerk
- (adaptives) Ratelimit
- Auswertung und Kombination der Symbole um Aktionen zu forcieren oder weitere Symbole (für die weitere Verarbeitung) zu erzeugen
- Signieren der Mail mit DKIM oder ARC
- Ggf. Auslösen von Greylisting, Lernen der Mail via Bayes und/oder Fuzzy
- Update der Datenbanken für Ratelimit und Reputation
- Triggern von Notifications - z.B. Mail an den Admin
- Eingebaute Redundanz und Failover mittels der Upstreams Funktion

Greylisting im Rspamd

- früher sollte Greylisting die SMTP-Konformität des einliefernden Servers beweisen
  - Ein SMTP Server muss Queuing beherrschen und bei einem temporären Feher (4xx) die Zustellung erneut versuchen
  - das konnte die Implementierung früherer Bot Netze (2005-2008) aber nicht
  - damals eine wunderbar wirksame Methode gegen Bot-Spam
- die alte Idee ist heute nicht / kaum mehr relevant oder sogar nicht hilfreich
- Aber Greylisting verzögert eine Mail typischerweise ca. 5 Minuten
- In 5 Minuten kann in einer RBL oder eigener Reputation genauso wie bei der Fuzzy Texterkennung viel passieren
- Also: heute verzögern wir eine Mail mittels Greylisting um ggf. kurze Zeit später die Mail besser bewerten zu können.
- Rspamd verzögert eine Mail meist auf Grund des Scores und merkt sich dabei auch den Body-Hash als Wiedererkennungswert (IP ist für das Greylisting meist irrelevant)

Rspamd - AI Threat Detection

- Entzaubern wir die AI/KI Buzzwords mal ein bisschen
- Rspamd sammelt eindeutige Ergebnisse seiner eigenen Scans getrennt nach HAM/SPAM
- Wenn eine bestimmte Anzahl an Reports zusammen gekommen ist, wird dieses Set einem Neuronalen Netzwerk übergeben
- Dieses versucht über mehrere Wiederholungen typische Merkmale der Spam und der Ham Mails zu erkennen
- Das angelernte Set wird danach zur Erkennung genutzt und neue Reports für das nächste Set gesammelt
- Das zu lernende Set kann dem Rspamd auch explizit bereit gestellt
- Es können mehrere Sets zum Einsatz kommen

Rspamd - Log

Mär 11 12:26:45 srv.example.com rspamd[4137084]: <1190ef>; task; rspamd_task_write_log: id: <q1aWo8J54nOoq1aWo8J54nOoq1aWo8J54nOo@iaeti.org>, qid: <E9972200407>, mta: 127.0.0.1 (localhost), ip: 198.251.80.254 (mail10.iaeti.org), from: <Valee@iaeti.org>, subject: "h mustermann, Erektionsprobleme?", (default: T (reject): [36.13/15.00]

- [**HS_RS_BAD_FAKE_SHOP**(12.00){},**RBL_AX_COMB_BLACK_HEUR**(6.00){198.251.80.254:from;},**BAYES_SPAM**(5.07) {99.93%;},**SCHAALIT_URI_903**(5.00){},**RCVD_UNAUTH_PBL**(2.00){},**R_DKIM_ALLOW**(1.52){iaeti.org:s=root;},**IP_REPUTATION_SPAM**(1.22){asn: 53667(0.40), country: US(0.01), ip: 198.251.80.254(0.00);},**SUBJECT_ENDS_QUESTION**(1.00){},**MX_INVALID**(0.50){},**DKIM_REPUTATION**(0.49) {0.99933343079923;},**SPF_REPUTATION_SPAM**(0.49){0.99933343079923;},**MIME_HTML_ONLY**(0.20){},**LOCAL_FUZZY_AUTOLEARN**(0.11){type 2 (weight: 2);},**BAD_REP_POLICIES**(0.10){},**LOCAL_RCPT**(0.10){example.com;},**NON_LOCAL_IP**(0.10){198.251.80.254;},**SPAMD**(0.10) {HTML_MESSAGE;MIME_HTML_ONLY;SPF_HELO_NONE;},VALID_RCPT(0.10){h.mustermann@example.com;},**ARC_NA**(0.00){},**ARC_SIGNED**(0.00) {example.com:s=arc:i=1;},**ASN**(0.00){asn:53667, ipnet:198.251.80.0/24, country:US;},**DEFAULT_INCOMING**(0.00){},**DKIM_TRACE**(0.00){iaeti.org: +;},**DMARC_POLICY_ALLOW**(0.00){iaeti.org;reject;},**FROM_EQ_ENVFROM**(0.00){},**FROM_HAS_DN**(0.00){},**GROUP_RBL_FROM_REJECT**(0.00) {},**HAS_URL**(0.00){},**INCOMING**(0.00){},**IXHASH_TEST**(0.00){38e264a10145a17e1488c4e276486a90;},**MID_RHS_MATCH_FROM**(0.00) {},**MIME_TRACE**(0.00){0:~;},**PROXY_INFO**(0.00){E9972200407;srv.example.com;1190ef;},**RCPT_COUNT_ONE**(0.00){1;},**RCPT_DN_IN_SUBJECT**(0.00) {dn mime: h mustermann;},**RCPT_USER_IN_SUBJECT**(0.00){rcpt mime user: h mustermann;rcpt smtp user: h mustermann;},**RCVD_AX_COMB_GEN_RDNS**(0.00){82.211.222.158:received;},**RCVD_COUNT_ONE**(0.00){1;},**RCVD_SH_ZEN_PBL**(0.00) {82.211.222.158:received;},**RCVD_TLS_LAST**(0.00){},**R_SPF_ALLOW**(0.00){+ip4:198.251.80.0/24;},**TO_DN_ALL**(0.00) {},**TO_MATCH_ENVRCPT_ALL**(0.00){},**UID_ROUNDTRIP_VERIFY_FAIL**(0.00){no_hdr;}])
- , len: 2605, time: 2552.095ms, dns req: 69, digest: <e2ac0d9e945a1b6d643c992836c9025e>, rcpts: <h.mustermann@example.com>, mime_rcpts: <h.mustermann@example.com>, settings_id: default_in

Rspamd - Selectors

- mittels Selectors kann man ohne (richtige) Programmierung jeden erdenklichen Wert aus einer E-Mail bzw. einem Scan-Vorgang abfragen
  - *liefert die ersten 16 Zeichen des Subjects*

    header('Subject').substring(1, 16)

  - *liefert die SHA256 Anhänge aller Anhänge*

    attachments(hex,sha256)

  - diese Selector Variablen können nun in vielen Plugins genutzt werden
    - RBL
    - Multimap
    - Settings
    - Force Actions (Reject Messages)
    - Reputation
    - Ratelimit

Rspamd - Lua Scripting

- überall wo mensch mit den vorhandenen Plugins oder mit den vordefinierten Selectors an Grenzen kommt, läßt sich Rspamd mittels Lua Funktionen oder ganzen Lua Plugins erweitern
- Rspamd bietet bereits eine umfängliche Lua-API, die viele benötigte Funktionen bereits bereit stellen (*task:get_recipients()*)
- eine Erweiterung für Rspamd ist also mit wenigen Zeilen Code geschrieben
- auch weitere Selectors lassen sich mit ein paar Zeilen Lua schnell hinzufügen

Wo haben wir den Rspamd erweitert?

- Skripte um spezifischen Spam zu erkennen
  - verschiedenste Header Checks
  - encrypted Zip + Passwort etc
- Anomalie (Fraud) Erkennung (vor allem ausgehend)
- Gefälschte Absender (Envelope, Header, Display Name etc.)
- User Profile (neben den Settings) - Anwendung von Policies
- Wiedererkennen bereits gescannter Mails (mit gespeicherten Metadaten)
- verschiedene kleine Patches für die offiziellen Plugins
- dynamisches (per-recipient) Routing in Postfix
- Xspct - Database - LDAP/SQL/x Anbindung via HTTP Python Daemon
- Xspct - Postfix - Policy-Daemon und TCP-Lookup Table um Rspamd schon früher / noch später in Postfix einbinden zu können

Rspamd und Sandboxing mit PeekabooAV

- PeekabooAV verhält sich wie ein Antivirus-Scanner und wurde ursprünglich für Amavis entwickelt
- neben weiteren Analysen wie Cortex oder Oletools kann PeekabooAV ein File in einer (Open-Source) Sandbox wie Cuckoo oder CAPEv2 starten lassen
- das Ergebnis der Analysen wird durch PeekabooAV selbst bewertet und entsprechend der eigenen Rules als *unauffällig* oder *böse* eingestuft

Rspamd und Sandboxing mit PeekabooAV

- Rspamd ist auf Geschwindigkeit optimiert und möchte innerhalb von wenigen Sekunden einen Scan abschließen
- Wir möchten bei eingehenden Mails von außen alle Prüfungen Pre-Queue machen - das heißt vor der Annahme der Mail
- Sandboxing und andere intensive Analysen benötigen aber vor allem Zeit
- Passt irgendwie nicht zusammen
- Aber wir können uns mit einem temporären Reject ja ca. 5 Minuten Zeit verschaffen
- Danach sind die meisten Analysen fertig

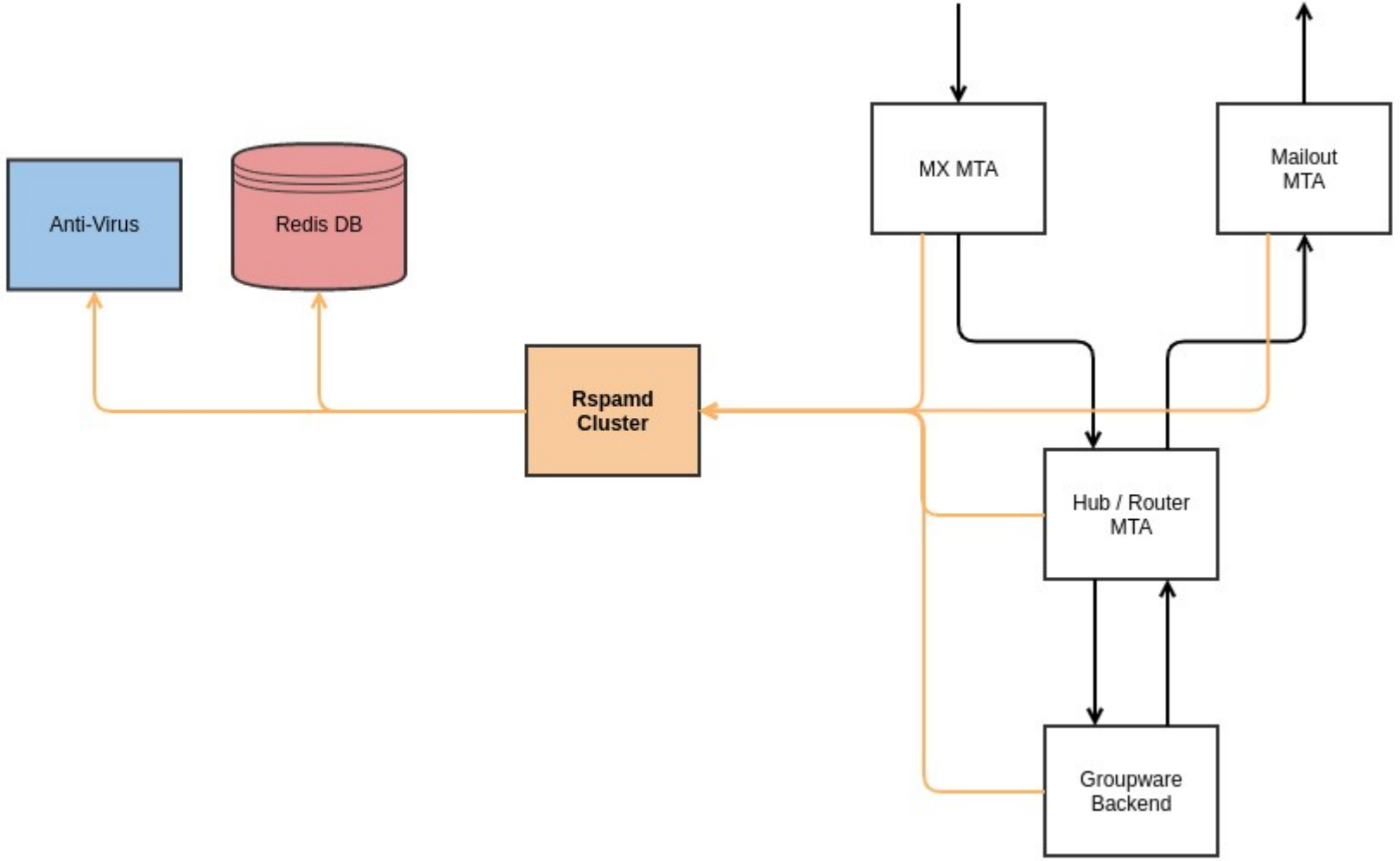
Rspamd und Sandboxing mit PeekabooAV

- PeekabooAV bekam rechtzeitig für unser Projekt eine HTTP Rest-API
- Wir entscheiden im Rspamd ob eine Mail im Sandboxing analysiert werden soll, senden den Anhang an PeekabooAV und merken uns die Scan-ID
- 2-3s (also noch während des Scans) fragen wir schon einmal nach dem Ergebnis - das kann bei einem Whitelisting dann schon vorliegen
- Wenn der Report noch nicht vor liegt, rejecten wir die Mail temporär
- Wenn die Mail nach 5 Minuten noch einmal eingeliefert wird, fragen wir nochmals bei PeekabooAV nach Report
- das Prinzip funktioniert super - aber uns fehlte noch etwas ...

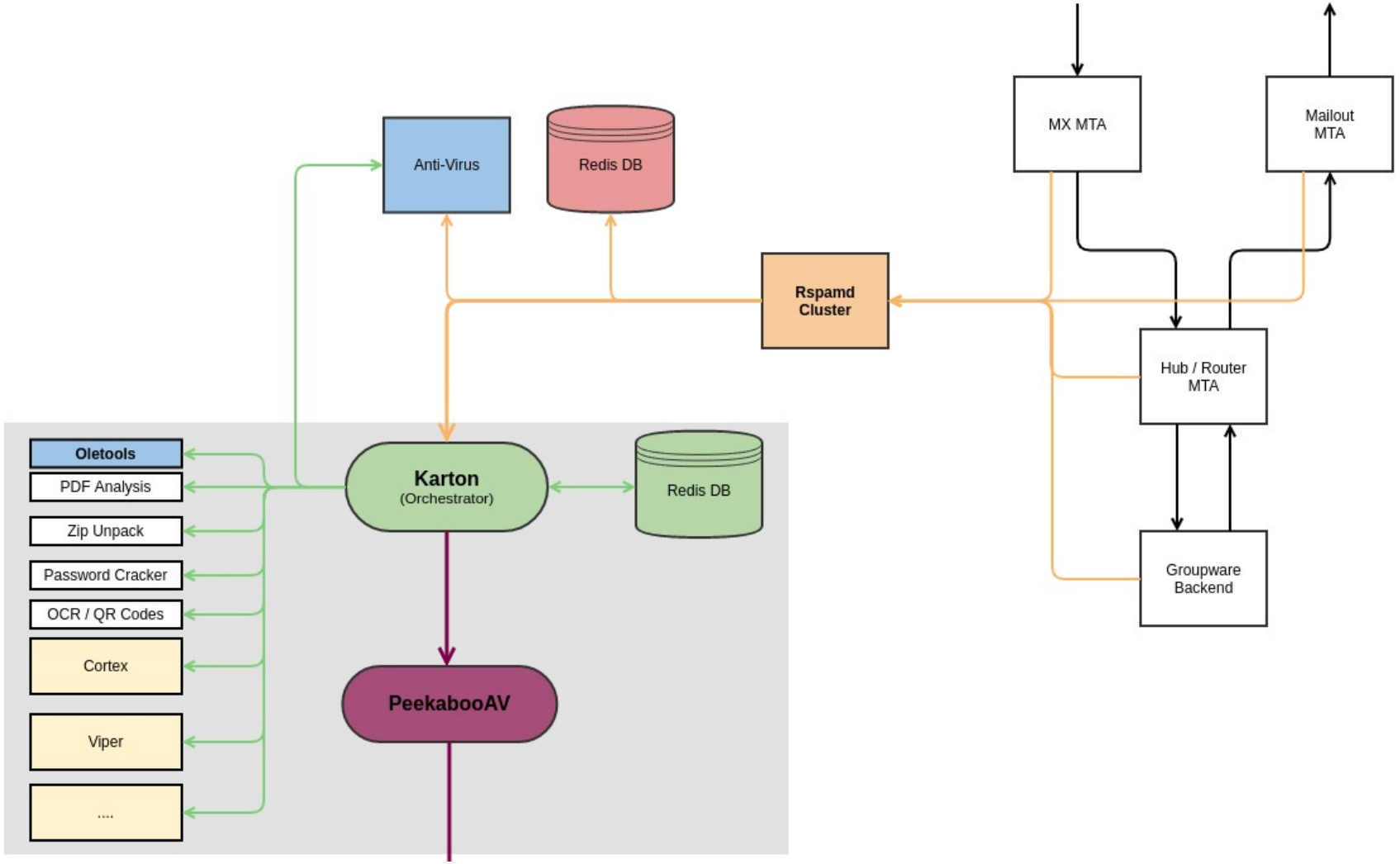
Rspamd + Karton + Expander + Peekaboo (Sandboxing)

- weder Rspamd noch PeekabooAV möchten Zips auspacken
- wir benötigen also noch eine entpackende Instanz dazwischen
- hier haben wir die Malware Analyse Plattform **Karton** vom CERT Polen gefunden, dass hier perfekt in das Konzept passt
  - Karton nimmt Files entgegen und schaut ob es für diesen Dateityp einen Interessenten gibt
  - bei einem Zip wäre das die Erweiterung *Extractor* - diese entpackt die Mail und schickt den Inhalt des Archivs wieder an Karton
  - für die entpackten Dateien interessiert sich nun PeekabooAV bekommt die Files zur Analyse übermittelt
- Unsere Kollegen von PeekabooAV haben alle notwendigen Plugins und Erweiterungen für Karton entwickelt und als **Expander** zu einem Paket zusammen gefasst
- Damit haben wir einen großartigen Ansatz um neben den schnellen Analysen im Rspamd, tiefgreifende Analysen in einem sehr flexiblen und erweiterbaren 2nd Stage Scanner zu machen - OHNE die Mail vorher angenommen zu haben
- Auch ohne den Einsatz von Sandbox Maschinen bietet dieser Ansatz einen großen Vorteil bei File Analysen
- *dieser Ansatz läuft in einem Pilotprojekt seit vielen Monaten produktiv*

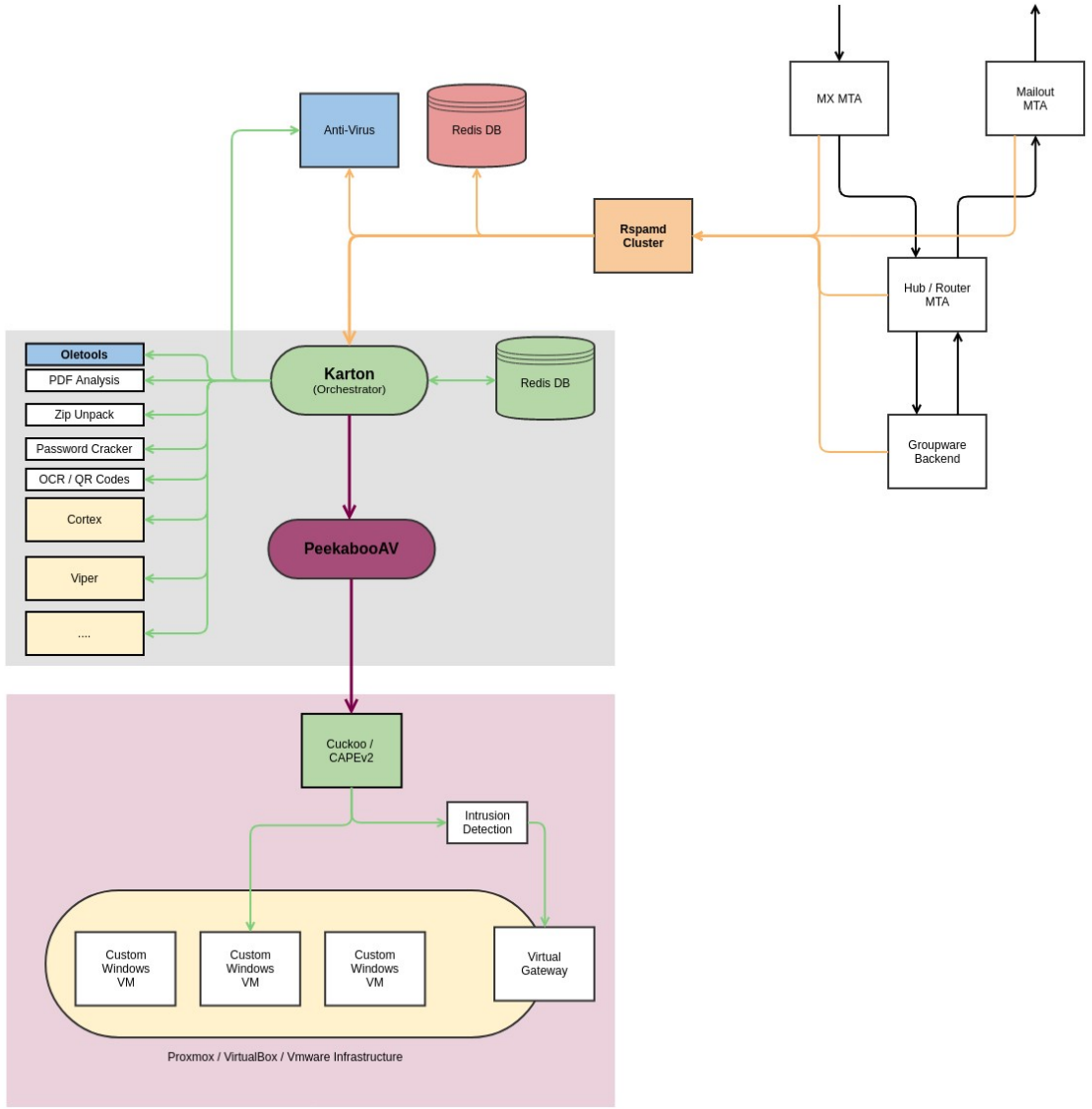
Rspamd + Karton + Expander + Peekaboo



Rspamd + Karton + Expander + Peekaboo



Rspamd + Karton + Expander + Peekaboo



Status Rspamd + Karton + Expander + Peekaboo

- erste Version der Rspamd Erweiterung ist fertig aber noch nicht Upstream integriert
- Beta Release der Karton Erweiterung *Expander* ist auf Github/PyPI verfügbar

  https://github.com/science-computing/expander
- PeekabooAV selbst ist stable und seit vielen Jahren produktiv im Einsatz

  https://github.com/scVENUS/PeekabooAV
- eine Herausforderung ist aber die fehlende bzw divergierte Weiterentwicklung des Sandboxing Tools *Cuckoo* dar
  - das Original-Projekt wird nicht mehr entwickelt
  - es gibt verschiedene Forks, die auch mit neuen Windows Versionen besser umgehen können
  - diese benötigen aber noch Anpassungen um als vollwertiger *Cuckoo* Ersatz zum Einsatz kommen zu können

Buzzword Bingo #2

- Adaptive Ratelimiting
- Advanced Anomaly Detection
- Advanced Malware Protection and Threat Grid
- Advanced Multi-layer Attack Vector Detection
- Advanced Multi-layer Malware Detection
- Advanced Threat Protection
- Artificial Intelligence (AI) Spam Detection
- BEC and CEO Fraud Detection
- Cloud-driven Reputation
- Content Disarm and Reconstruction
- Domain Fraud Protection

Buzzword Bingo #2

- Email Data Loss Prevention
- Identity-Based Encryption (IBE)
- Impersonation Analysis
- Local and Cloud Sandboxing
- Mailbox Safeguard
- Recipient dependend Transport Layer Security and Encryption
- SIEM Vector Export
- Typosquatting Detection
- URL Click Protection
- Web Interaction Tracking
- Secure Message Delivery
- Virus Outbreak Protection

Advanced Threat Protection

- *Alias: Advanced Anomaly Detection, Advanced Malware Protection and Threat Grid, Advanced Multi-layer Attack Vector Detection, Advanced Multi-layer Malware Detection, Virus Outbreak Protection*

  Erkennung schädlicher E-Mail Inhalte oder Anhängen

  Mehrstufig: Dateiendung, Hash, Anti-Virus, File-Analysis, Cloud-Abfrage, ggf. Sandbox

- Rspamd: Multimap, Antivirus, RBL, Oletools/PDF, PeekabooAV

Artificial Intelligence (AI) Spam Detection

- *Alias: Cloud-driven Reputation*

  Erkenntnisse aus dem Lernen verschiedener Sets in neuralen Netzwerken
  Abruf oder Abfrage der Daten aus der Hersteller Cloud

- Rspamd: local Neural Network, local und remote Fuzzy

Impersonation Analysis

- *Alias: BEC and CEO Fraud Detection, Domain Fraud Protection*

  Erkennung gefälschter Absender (bei der eigenen Domain)

  Angreifer versuchen mit gespooften Envelope, Header-From oder Display Name einen das vertrauen des Empfängers zu erlangen um diesen zu den gewünschten Handlungen zu bewegen

- Rspamd: Built-In Funktionen, Multimap + Composites (meine Domain von außen), DKIM/DMARC, Lua-Funktionen, ausgehend ggf. User-Profil Funktionen

Mailbox Safeguard

- *Alias: Clawback*

  nachträgliches Löschen von Mails aus den Mailboxen der User

- Rspamd: Das könnte mit den Daten vom Rspamd natürlich gebaut werden, wir glauben aber nicht, dass die Methode effektiv ist. Besser wäre ein Ansatz eine Mail vor der Anzeige beim User noch einmal zu prüfen.

Email Data Loss Prevention

hier macht jeder Hersteller ein bisschen was anderes
typischerweise möchte man verhindern, dass bestimmte Daten per Mail versand werden
mögliche Ansätze: Dateinamen, -typen, Hashes, bestimmte Triggerbegriffe

- Rspamd: eigentlich wie eingehend - Multimap + Composites

Secure Message Delivery

- *Alias: Recipient dependend Transport Layer Security and Encryption*

  DANE, MTA-TLS, forciertes TLS

Identity-Based Encryption (IBE)

Mail wird nicht versendet, sondern über ein Webportal bereit gestellt

kann mit verschiedenen Diensten umngesetzt werden

Typosquatting Detection

Erkennung leicht anders geschriebener Versionen bekannter (eigener) Domains

Rspamd: Lua - Levenshtein Funktion

Sind kommerzielle Mail-Security Appliances besser?

- scheinen als fertige (Black-) Box sofort einsatzbereit
- viele durchaus sehr gute Mehrwerte der kommerziellen Anbieter basieren auf Daten aus ihrer Cloud
- Der Zusatzwert ist (auch) aus den Daten der Kunden entstanden
- Jeder der die Cloud abfragt, liefert auch neue Daten
- Daher können die Anbieter ohne weiteres behaupten jeden Tag hunderte Milliarden an Mails zu analysieren
- Sie haben alle meist ein fancy Webinterface
- Open-Source Sandbox-Lösungen steht noch vor ein paar Herausforderungen

Wir stehen den kommerziellen Anbietern also technisch in fast nichts nach, uns fehlen aber die globalen Analysedaten und damit die umfangreichen Reputationsdaten

externe Reputationsdienste im Rspamd

- wir bauen im Rspamd unsere eigene Reputation für viele Indikatoren auf
- wir machen eine Textanalyse mit Fuzzy und Bayes und bauen uns daraus eine Datenbank auf

- das ist wichtig und gut, weil es genau zu unserer Umgebung passt

- trotzdem sind weiterhin auch externe Reputationen wie z.B. RBL's sinnvoll
- Alternativ: Text-Listen, HTTP-API's etc
- aber im Gegensatz zu früher nutzen wir RBLs nicht mehr zum sofortigen Abweisen einer Mail, sondern es ist einer von vielen Indikatoren für eine Mail

externe Reputationsdienste am Beispiel von Spamhaus

- IP-Reputation
  - SBL - Spamhaus Block List
  - XBL - Exploits Block List
  - PBL - Policy Block List - Endkunden (DSL) Anschlüsse
  - AuthBL - Auth Blocklist - IP von denen Brute-Force Attacken ausgingen
- Domain Reputation
  - DBL - Domain Block List
  - ZRD - Zero Reputation Domain - hier werden frische Domains gelistet
- Hash Reputation
  - Mail Adressen - bekannte Spam-Mailadressen aus From oder Reply-To
  - Bitcoin Adressen - aus Spams bekannte Bitcoin/ETC ... Adressen
  - File Hashes - Hashes schadhafte Attachments
- ZRD / HBL setzen eine kommerzielle Subscription voraus

Rspamd Fuzzy

- Rspamd Fuzzy ist ein Shingles Algorithmus, der die Schnittmenge zweier Texte berechnen kann
- Anders als Bayes ist es keine statistische Analyse
- mit dem Fuzzy Algorithmus im Rspamd und der Art wie die Fuzzy Hashes gelernt werden können, ist es möglich eine eigene als auch gemeinsame gefütterte Datenbank aufzubauen
- Fuzzy Hashes werden zusammen mit einer Gewichtung gespeichert

  Verwendung des Hashes erst ab einem bestimmten Schwellwert
- Absicherung über Public/Private Keys
- geteilte Fuzzy Datenbanken sind dabei für verschiedenste Konstellationen denkbar

  Public - Community - Crowd

  geschlossene Gruppe

  kommerzielle Angebote

Interesse / Also was fehlt uns noch?

- Webinterface
- Zeit Patches upstream zu bringen
- Karton / Peekaboo Engagement

Fazit

- Wir können auch die ganzen Enterprise Buzzword Features umsetzen
- Wir können abgesicherte Mailcluster voll redundant und lastverteilend rein mit Open-Source Mitteln aufbauen
- Die Komponenten müssen/können/dürfen aber selbst zusammen gestellt werden
- Die ganzen Cloud Abfragen der kommerziellen Hersteller können wir nicht direkt ersetzen, aber wir können uns ähnliche Daten selbst erstellen, wenn wir genügend Mails / Teilnehmer haben

heinlein

Fragen und Diskussionen

Linux höchstpersönlich.

Einschränkungen bei Postfix

- Wir können Postfix über das Milter Protokoll keine detailierten Routing-Anweisungen mitgeben
- Aber Adressen umschreiben oder Header setzen
- via Header können wir mittels dem FILTER Command in den header_checks die komplette E-Mail umleiten - das ist nicht immer sinnvoll
- Daher tricksten wir ein bisschen und hängen während der internen Verarbeitung einen Domain-Suffix an die Adressaten, den wir später wieder im Postfix entfernen
- Dieser Suffix kann auch einen eindeutigen Hash beinhalten, über welchen wir später beim Versand das konkrete Ziel eines Adressaten beim Rspamd abfragen

Einschränkungen beim Rspamd

- Rspamd ist eigentlich eine Rest-API, kann mit Teilinformationen nicht so viel anfangen und beginnt seine Arbeit erst wenn die Mail vollständig übertragen wurde (Nach dem ./EOM - die Mail ist trotzdem noch nicht angenommen)
- Daher Rspamd nicht schon bei den einzelnen SMTP-Empfängern arbeiten (RCPT TO:)
- Hier helfen wir uns mit einem kleinen Policy Service für den Postfix, der jeden Empfänger schon einmal gesondert mittels einer selbst generierten *Fake*-Mail anfragt
- Rspamd kann über das Milter-Protokoll keine Änderungen am Body zurück geben
- - technisch kann Rspamd sogar ziemlich gut den Body anpassen, aber die HTTP Antwort vom Rspamd müsste an anderer Stelle verarbeitet werden
- Rspamd kann nicht mit klassischen Datenbanken wie SQL oder LDAP umgehen - blockierende Operationen passen nicht in das Rspamd Konzept
- Wir haben einen kleinen Daemon, der Datensätze für den Rspamd aus dem LDAP/SQL holt und Rspamd via Rest-API bereit stellt

Die Heinlein-Gruppe: Gemeinsam für digitale Souveränität

Heinlein Support

- Akademie: Für die oberen 10% des Wissens – unsere Linux-Schulungen für IT-Experten.
- Consulting: Security- und Strategieberatung, Projektumsetzung und umfassender Support für IT-Administratoren
- Services: SLA-Verträge, Hosting und Lizenzen als Unterstützung & Absicherung Ihrer kritischen IT-Infrastruktur

Weitere Marken

- mailbox.org: E-Mail, Online-Office, Cloud-Speicher und Videokonferenzen nach neuesten Sicherheitsstandards und mit grüner Energie.
- OpenTalk: Videocalls, wie sie sein sollten – mit unserer sicheren, benutzerfreundlichen und skalierbaren Videokonferenz für Behörden, Provider, Unternehmen und Schulen.

CompetenceCall

Das Backup für Ihre Server-Administration.

Nutzen Sie unsere SLA-Verträge und sichern Sie sich den 24/7-Support unserer Linux-Consultans.

- Kontimuierliche Absicherung mit garantierten Reaktionszeiten und festen SLAs
- Rückendeckung im Notfall: mindestens LPIC-2 zertifizierte Profis mit jahrelanger, täglicher' Admin-Erfahrung
- Projektunterstützung: maßgeschneiderte Lösungen, die Flexibilität, Sicherheit, Administrierbarkeit und Hochverfügbarkeit vereinen
- Services: Performanceanalyse, Serverhärtung, Netzwerkanalyse, Konfigurationshilfe, Datenrestaurierung

Werde Teil des Teams

- Du bist neugierig, voller Tatendrang und überzeugt von Linux, Open Source und sicherer, freier Kommunikation?

- Wir freuen uns über Unterstützung im Team:
  www.heinlein-support.de/jobs

Bleiben wir im Kontakt

Carsten Rosenberg

Tel. +49 30 40 50 51-46
c.rosenberg@heinlein-support.de

Heinlein Support GmbH
Schwedter Straße 8/9 | 10119 Berlin
www.heinlein-support.de

Bleiben wir im Kontakt

Manu Zurmühl

Tel. +49 30 40 50 51-51
m.zurmuehl@heinlein-support.de

Heinlein Support GmbH
Schwedter Straße 8/9 | 10119 Berlin
www.heinlein-support.de