Stay Open.

Open-Xchange Abuse Shield

Major Release v2.0.0

Product Guide

1 OX Abuse Shield

1.1 Intention of this document

This document provides an overview of the improvements and other changes that come with this major release of OX Abuse Shield, v2.0.

The purpose of this document is to inform Open-Xchange customers and partners about the major changes that have been made in this release.

1.2 Product Re-Naming with new Version

Open-Xchange decided to re-name the Product from Dovecot Anti-Abuse Shield to OX Abuse Shield. This name change corresponds to the ability to use OX Abuse Shield in different environments that do not include Dovecot or OX App Suite, for example with other mail systems, web portals etc.

1.3 OX Abuse Shield in General

OX Abuse works with both Dovecot Pro and OX App Suite, or any third-party software via a REST API, as a component to protect against login/authentication abuse.

OX Abuse Shield runs on a cluster of servers and integrates with both OX App Suite and Dovecot to detect abuse, brute force attacks and also to enforce common authentication/authorization policies across the platform.

1.4 Key Features

- Replicated/clustered architecture – Login reports are shared between all the servers in a cluster so there is a single view of abuse

- Scriptable Policy Language – Using the Lua language, the functionality of the daemon can be extended to record and protect against a large variety of abusive behavior, as well as implement specific customer policies.

- DNS Lookup Feature – For looking up IPs or domains in blacklists

- GepIP Lookup Feature – GeoIP lookups can be made and incorporated into policy decisions.
- Rate limiting and Tarpitting – Extremely flexible, these can be enabled and enforced based on IP address, login name, geoip location, time windows, etc.
- Flexible In-Memory Statistics Database – A versatile and extensible in-memory database is used to store statistics information about abuse over time periods from a few minutes to many hours.
- Integration with Customer Authentication/Authorization Systems – Customers can use the open HTTP REST API to benefit from the protection of the anti-abuse daemon in their own authentication/authorization systems.
- Admin Console – To retrieve statistics and query server state
- Persistent Replicated Blacklist – Configurable via a REST API or the Lua policy engine, supports auto-expiry of entries, replication between all cluster nodes, and optionally uses a Redis DB for persistence.
- Webhooks – Integrate Anti-Abuse Shield with other systems using webhooks to send events.

A more detailed overview of Dovecot Anti-Abuse Shield can be found at the

https://software.open-xchange.com/products/weakforced/doc/OX_Whitepaper_OX_Abuse_Shield_2_0_0.pdf

1.5 The Goal of OX Abuse Shield

The goal of OX Abuse Shield is to detect brute forcing of passwords across many servers, services and instances, as well as enforce policy for authentication and authorization. In order to support the real world, brute force detection policy can be tailored to deal with "bulk, but legitimate" users of your service, as well as botnet-wide slow-scans of passwords.

Here is how it works:

- Report successful logins via JSON http-api
- Report unsuccessful logins via JSON http-api
- Query if a login should be allowed to proceed, should be delayed, or ignored via JSON http-api
- OX App Suite and Dovecot's POP/IMAP server are pre-integrated with OX Abuse Shield. Other software can be integrated easily using the REST API

2 Key benefits of OX Abuse Shield v2.0

Open-Xchange is pleased to announce the release of OX Abuse Shield v2.0.

Keeping in line with Open-Xchange’s security strategy OX Abuse Shield v2.10 contains many enhancements designed specifically for end-users and end-user usability: usability that stretches beyond the OX App Suite user base.

2.1 New Features

New in version 2.0 is a new daemon to detect and alert users about suspicious logins, as well as provide information to abuse teams about potentially compromised user accounts and abusing IP addresses.

Additionally, the new version provides:

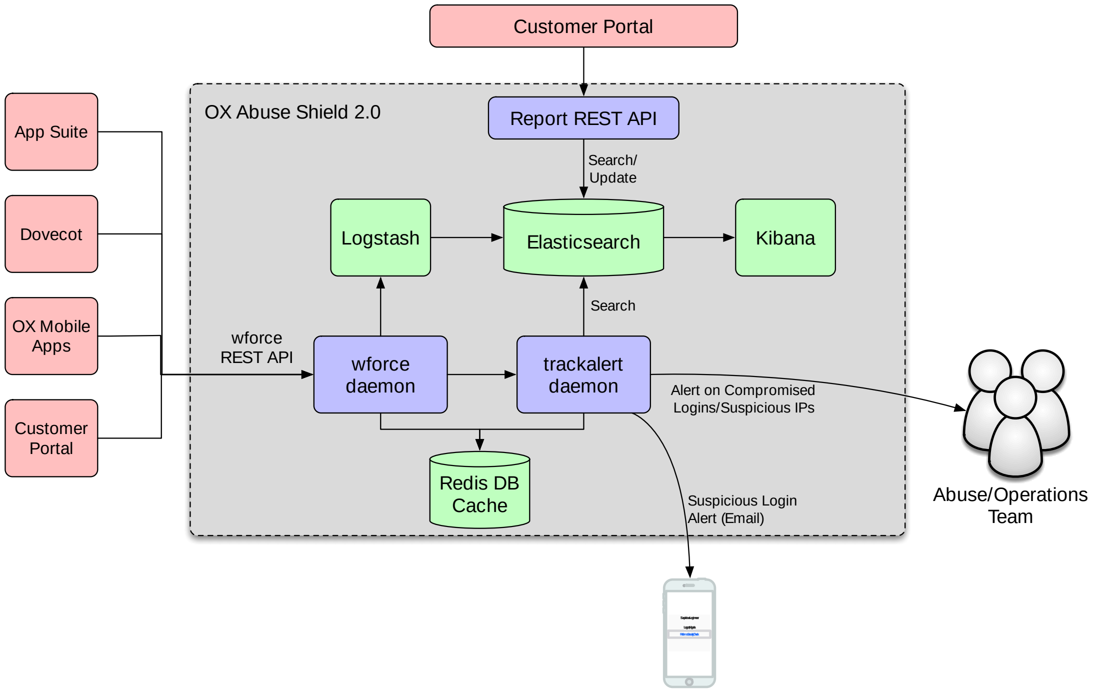
- Long-Term Storage of login data in Elasticsearch – Used for sophisticated anomaly detection features such as detecting suspicious logins.
- New “trackalert” daemon that integrates with Elasticsearch to detect anomalies.
- Suspicious Login Detection – Sending email alerts to end-users or webhooks to abuse/operations teams when logins are detected as suspicious due to anomalies from previous logins.
- Compromised Account Reports – Periodic reports, based on long-term data stored in Elasticsearch, sent via webhooks to abuse/operations teams about potentially compromised users and IP addresses abusing the system.

- Pre-Configured Kibana Reports and Dashboards – Including system-wide as well as per-IP and per-login forensic dashboards.
- Integration with LDAP for retrieving email alert addresses.
- Integration with ELK (Elasticsearch, Logstash, Kibana) stack.
- REST API providing access to data stored in Elasticsearch about previous logins, known devices etc. Allows applications to display login history and devices to end-users, as well as to delete devices that are no longer used.

3 In Detail – Benefits and new Enhancements

3.1 Overview

For an overview of the existing and new components and how they fit together, see the following diagram:



3.2 Long-Term Storage of Login Data in Elasticsearch

The new functionality is used for sophisticated anomaly detection features such as detecting suspicious logins. When a login report is received, the wforce daemon can be configured to send the report to both Elasticsearch and the new “trackalert” daemon. The trackalert daemon then queries Elasticsearch to determine if the login is suspicious, using data about previously known good logins, including the device used, the IP address and the country. If a login is determined to be suspicious, an alert can be sent; either directly to the user over Email; or using a webhook to an HTTP endpoint for processing by the operations team.

Additionally, regular reports can be run against the login data stored in Elasticsearch to determine IPs that are potentially abusing the system, and users whose accounts may be compromised.

3.3 New “trackalert” Daemon

The new trackalert daemon is similar to wforce, and similarly uses a Lua engine for configuration and policy. A standardized policy is provided, which can be customized, and which provides all the features described in this document.

3.4 Suspicious Login Detection

Sending email alerts to end-users or webhooks to abuse/operations teams when logins are detected as suspicious due to anomalies from previous logins.

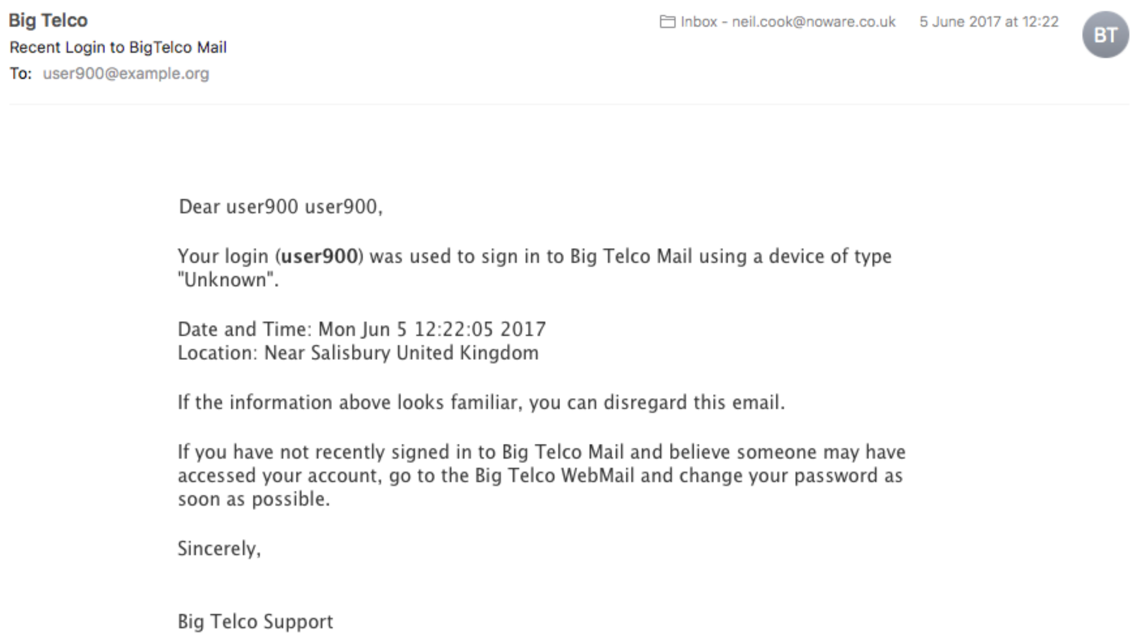
The new standardized policy provides the new behaviors:

Suspicious Login detection using IP Address, Country, and Device as the parameters to detect whether a particular login matches previous known good logins

- Integration with Elasticsearch and Redis for very powerful yet also extremely efficient searching of long-term login behavior

- Support for sending alerts via webhook or SMTP about suspicious logins

An example Email alert is shown below:



Big Telco
Recent Login to BigTelco Mail
To: user900@example.org

Inbox - neil.cook@noware.co.uk 5 June 2017 at 12:22

Dear user900 user900,

Your login (**user900**) was used to sign in to Big Telco Mail using a device of type "Unknown".

Date and Time: Mon Jun 5 12:22:05 2017
Location: Near Salisbury United Kingdom

If the information above looks familiar, you can disregard this email.

If you have not recently signed in to Big Telco Mail and believe someone may have accessed your account, go to the Big Telco WebMail and change your password as soon as possible.

Sincerely,

Big Telco Support

3.5 Compromised Account Reports

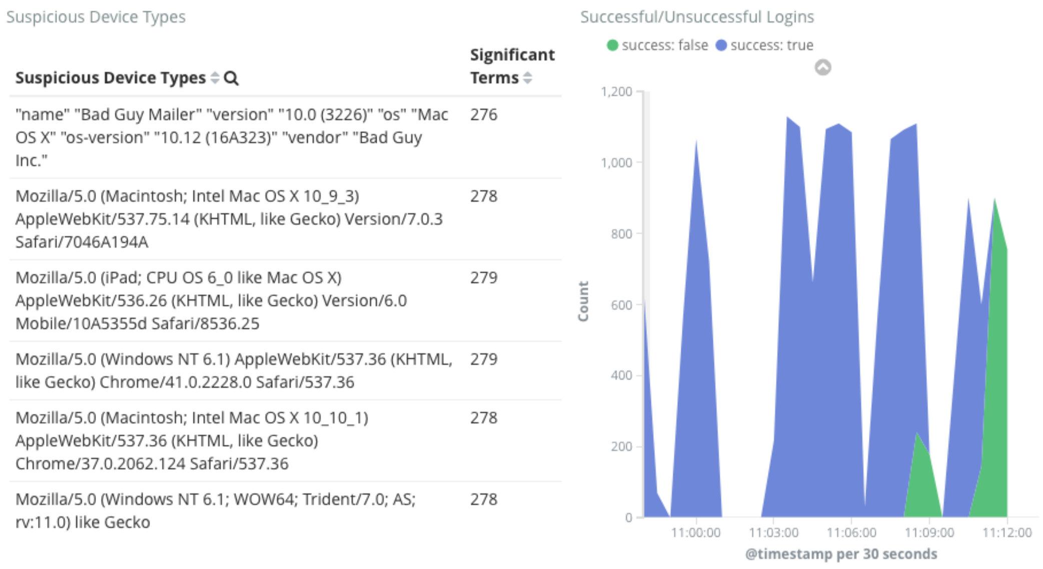
Periodic reports, based on long-term data stored in Elasticsearch, sent via webhooks to abuse/operations teams about potentially compromised users and IP addresses abusing the system.

Support for detecting compromised accounts and suspicious IPs and sending alerts via webhook on a periodic basis (e.g. once per day, every 6 hours etc.).

3.6 Pre-Configured Kibana Reports and Dashboards

Including system-wide as well as per-IP and per-login forensic dashboards.

Part of an example dashboard is shown below:



Suspicious Device Types

Suspicious Device Types	Significant Terms
"name" "Bad Guy Mailer" "version" "10.0 (3226)" "os" "Mac OS X" "os-version" "10.12 (16A323)" "vendor" "Bad Guy Inc."	276
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_9_3) AppleWebKit/537.75.14 (KHTML, like Gecko) Version/7.0.3 Safari/7046A194A	278
Mozilla/5.0 (iPad; CPU OS 6_0 like Mac OS X) AppleWebKit/536.26 (KHTML, like Gecko) Version/6.0 Mobile/10A5355d Safari/8536.25	279
Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2228.0 Safari/537.36	279
Mozilla/5.0 (Macintosh; Intel Mac OS X 10_10_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/37.0.2062.124 Safari/537.36	278
Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; AS; rv:11.0) like Gecko	278

Successful/Unsuccessful Logins
● success: false ● success: true
Count
1,200
1,000
800
600
400
200
0
11:00:00
11:03:00
11:06:00
11:09:00
11:12:00
@timestamp per 30 seconds

3.7 Integration with LDAP

Integration with LDAP is supported for looking up email address, first and last name etc., so that alert emails can be correctly addressed and personalized, and to prevent such emails from being mistaken for phishing messages.

3.8 Report REST API

The Report REST API is designed to allow integration of the Elasticsearch data into a web portal, by displaying for example last logins, known devices etc.

The REST API provides the following endpoints:

- /logins – Return information about logins matching the supplied criteria.
- /devices – Return information about devices matching the supplied criteria
- /logins/confirm – Confirm a login was valid (or invalid)
- /devices/forget – “Forget” a particular device (i.e. consider it a new device in the future).