
# TECHNOLOGY PARTNER PROGRAM

## USE CASE DOCUMENTATION

Author: Boldon James

# Contents

## Partner Information

| Partner information | |
|---|---|
| Date | November 11, 2019 |
| Partner Name | Boldon James |
| Web Site | https://www.boldonjames.com/ |
| Product Name | Classifier Foundation Suite |
| Partner Contact | Steve Cooper<br>(866)633 1116<br>Director – Product Technology & Partner Alliances<br>Steve.Cooper@BoldonJames.com |
| Support Contact | Support@BoldonJames.com<br>US: +1 (866)633 1116<br>UK: +44 (0) 3333 444 739 |
| Product Description | Boldon James data classification solutions helps you categorize, label and protect data. Advanced data classification fuses categorization with labelling to deliver a powerful data protection and governance punch. Enabling organizations to better manage and control their data, streamline operational performance and improve return on technology investment. |

## Use cases for integration into Palo Alto Networks Next-Generation Security Operating Platform

- Data Filtering with Custom Boldon James Classifier
    - Detect Classified Files crossing the NGFW
    - Alert/Block Files crossing the NGFW

## Palo Alto Networks Products for Integration

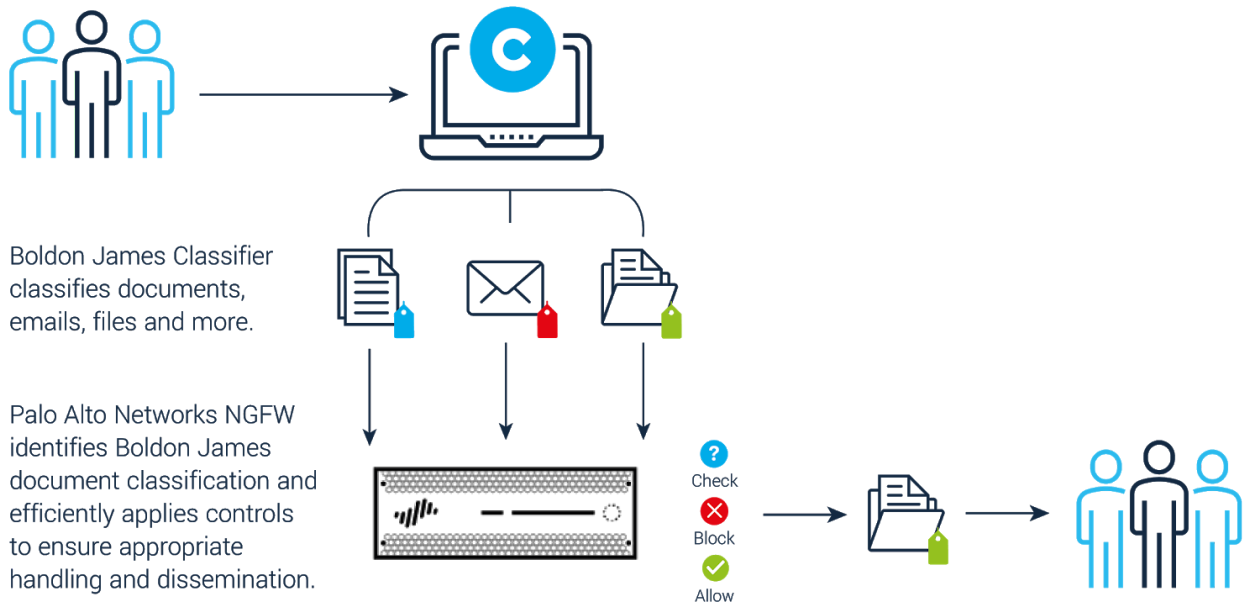
| Palo Alto Networks Product | Integration Status | Palo Alto Networks versions tested | Boldon James versions tested |
|---|---|---|---|
| AutoFocus | | | |
| Cortex XDR | | | |
| Cortex XDR Analytics | | | |

| | | | |
|---|---|---|---|
| MineMeld | | | |
| NGFW | Validated | PAN-OS 9.0 | Version 3.1 |
| Panorama | | | |
| Prisma Access | | | |
| Prisma Cloud | | | |
| Prisma SaaS | | | |
| Traps | | | |
| VM-Series | | | |
| WildFire | | | |
| Other | | | |

## Integration Benefits

- The combined solution between Palo Alto Networks NGFW and Boldon James allows granular detection of Classifier through NGFW Data Filtering
- Provide enforcement of Blocking/Alerting files passing via Network across the Palo Alto Networks NGFW

## Integration Diagram

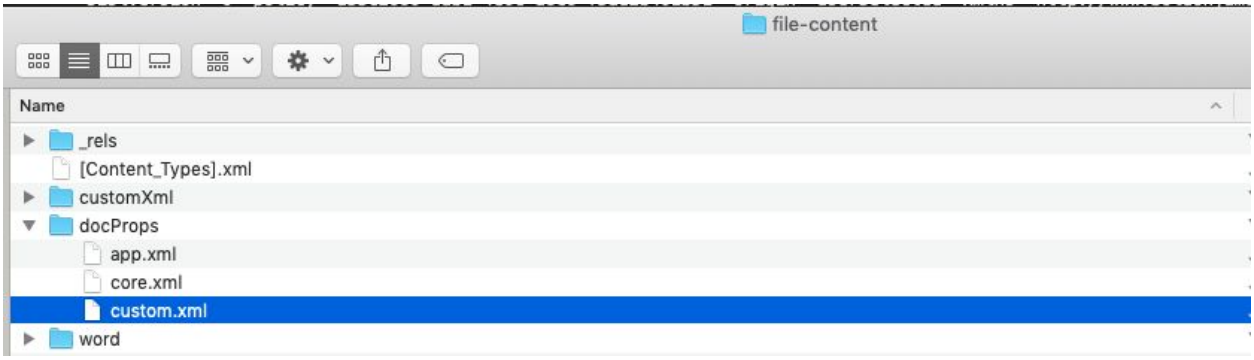


# Palo Alto Networks Configuration

## Patterns and Selection Strategy

The raw data can be viewed by unpacking one of your files with the Boldon James tagging, then locating the custom.xml under the docProps folder or viewing the file properties custom tab:

*Unzipped Word .docx with Boldon James Classifier*

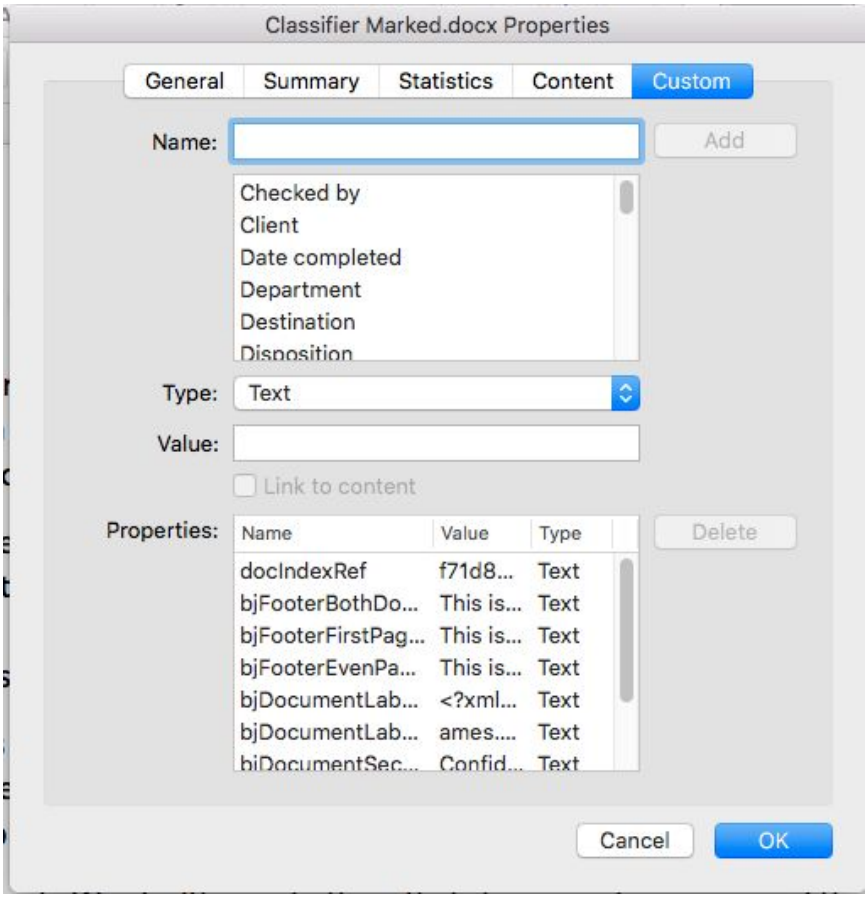



*Raw XML of custom.xml*

## File Property Pattern Type Method

Use the exposed Keywords field that is a Pattern Type under File Property.

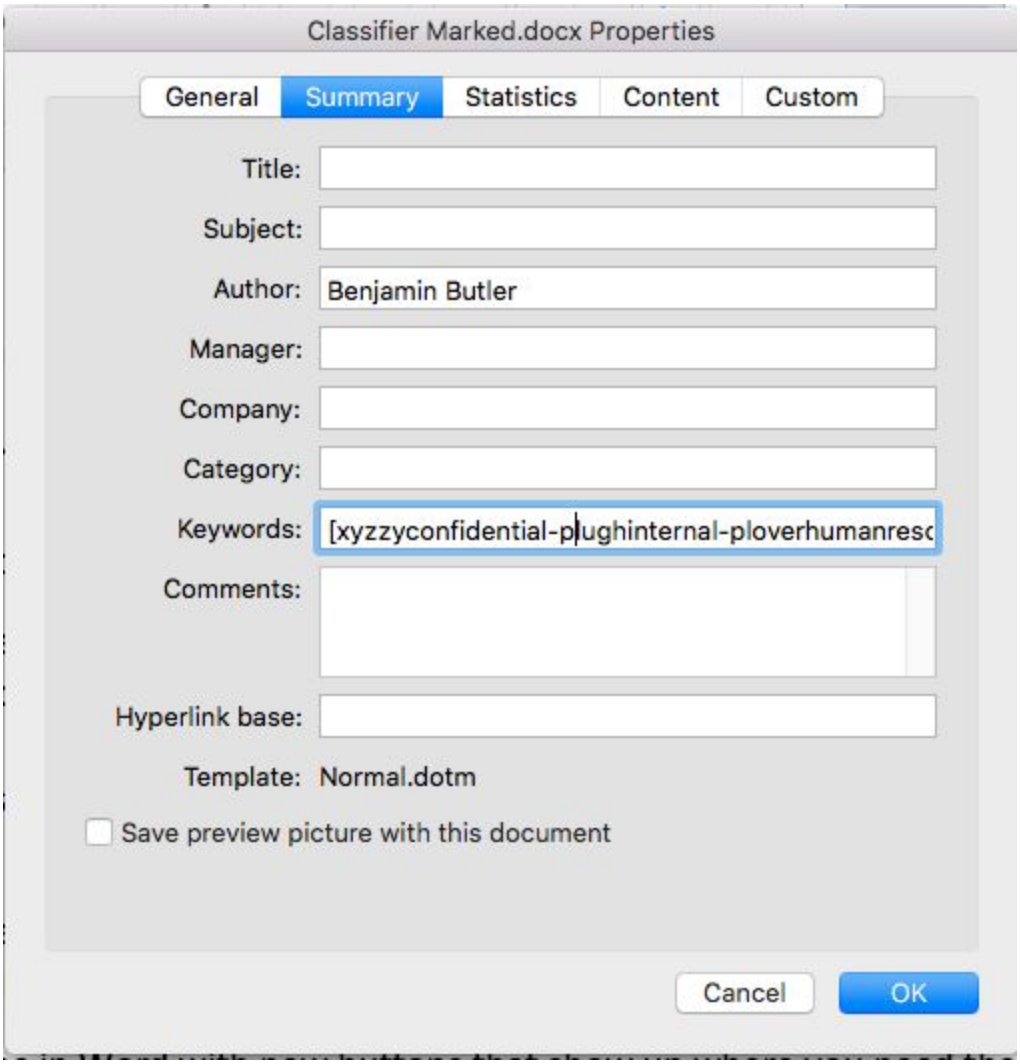
*Document Properties Custom Fields*



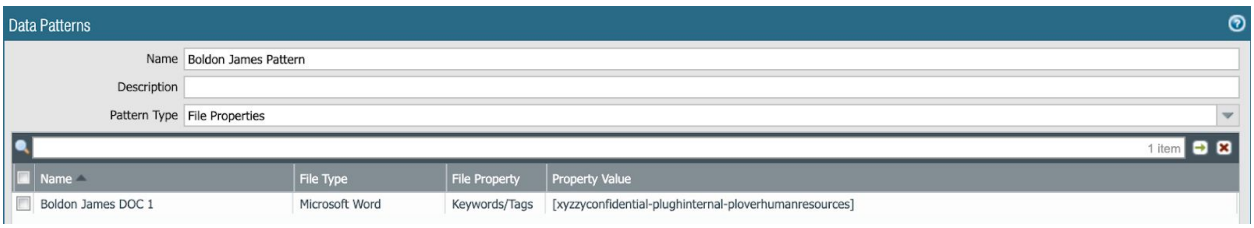
## File Property Pattern Type Method

Using the exposed Keywords field that is a Pattern Type under File Property.

*Document Properties Keyword Field*




*Data Patterns*




*Objects > Custom Objects > Data Pattern Profile > Pattern Type File Property*

## Regular Expression Pattern Type Method

You can choose to add context in several sections. Your file will be unique to your configuration, environment and so on, so use this as an example as a guide, but make sure the specifics are relevant to you.

*Example strings in custom.xml*

```
property fmtid="{D5CDD505-2E9C-101B-9397-08002B2CF9AE}" pid="3"
name="bjFooterBothDocProperty"><vt:lpwstr>This is a CONFIDENTIAL document
and restricted to INTERNAL staff members only._x000d_
DO NOT DISTRIBUTE OUTSIDE THE ORGANISATION.</vt:lpwstr></property>
```

There is a Key Value combo here to look at **name="bjFooterBothDocProperty"** and **"This is a CONFIDENTIAL"**.

*Focused string for our REGEX*

```
name="bjFooterBothDocProperty"><vt:lpwstr>This is a CONFIDENTIAL
```

We can create a data pattern on any of the keywords in the properties. The strategy used in this demonstration was a combination of unique patterns to eliminate false positives.

To match bjFooterBothDocProperty anywhere in the file we write, followed by CONFIDENTIAL anywhere after the first match:

**.*(bjFooterBothDocProperty).*(CONFIDENTIAL)**

**Note**: This is greedy in regex terms, meaning using * wildcards will cause a performance hit, so using something more specific is better. However the trade-off is the more specific, the greater the likelihood of False Negatives goes up.

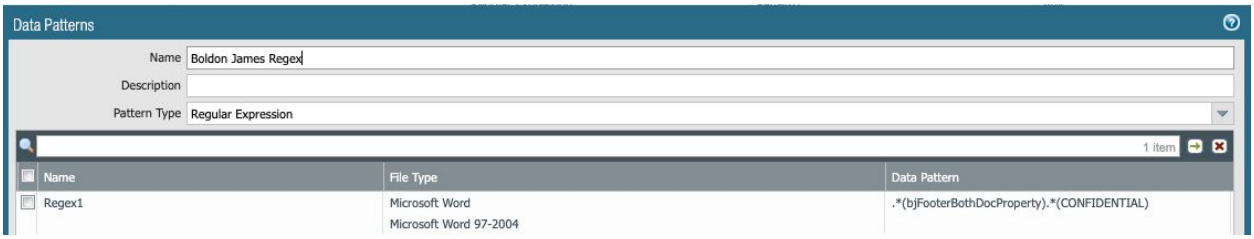
Also if we looked further within the XML, we could see the potential for at least 3 more examples to match on:

*Example Regex Patterns*

```
.*(bjFooterBothDocProperty).*(CONFIDENTIAL)
.*(bjFooterEvenPageDocProperty).*(CONFIDENTIAL)
.*(bjDocumentSecurityLabel).*(CONFIDENTIAL)
.*(bjDocumentLabelXML).*(CONFIDENTIAL)
```

This would look like the following on the NGFW:

*Data Pattern Examples*




*Objects > Custom Objects > Data Pattern Profile > Pattern Type Regular Expression*

### Filtering

Then we must add the Pattern to a Data Filtering Profile.

## Troubleshooting

- *Support:*
  - Support@BoldonJames.com
  - US: +1 (866)633 1116
  - UK: +44 (0) 3333 444 739
- *Boldon James is a member of* TSA Net