Mandiant

PUBLIC PREVIEWS FOR THREAT INTELLIGENCE API

This feature is released as a Public Preview.
Pre-GA products and features are available "as is" and might have limited support. For more information, please contact your TSC, your CSM, or go to **Support.**
(https://docs.mandiant.com/home/mandiant-support-cases)

You can download Mandiant YARA rules though our V4 API. The rules which are available to download from the API include the following:

- The content of the rule itself
- The malware family name which the rule is mapped to
- Other basic metadata like first and last updated time


```
{
    "created": "2023-12-06T18:18:15.843Z",
    "id": "signature--f26e2bf1-d6e0-5541-b465-cb22db249080",
    "malware": [
      {
        "id": "malware--1d7e3376-b176-5b17-8700-a1f3426b0df2",
        "name": "TRICKYBITS"
      }
    ],
    "modified": "2023-12-07T18:45:02.100Z",
    "name": "M_Hunting_Dropper_TRICKYBITS_1",
    "rule": "rule M_Hunting_Dropper_TRICKYBITS_1 {\n\tmeta:\n\t\tauthor = \"Mandiant\"\n\t\tmd5 = \"
9cb0aca5d284fc27b32104741e8c114b\"\n\t\tdate_created = \"2023-12-06\"\n\t\tdate_modified =
\"2023-12-06\"\n\t\trev = \"1\"\n\tstrings:\n\t\t$deccall_1 = { 48 85 C0 74 ?? 4C 8D 44 [2] 8B D6 48
8B C8 E8 [2] FF FF 48 8B D8 48 85 C0 74 ?? 8B 54 [2] 48 8B C8 E8 [2] FF FF 48 8B CB FF [3] 00 00 }
\n\t\t$deccall_2 = { 48 85 C0 74 ?? 4C 8D 44 [2] C7 44 [2] 00 00 00 00 8B D6 48 8B C8 E8 [2] FF FF 48
8B D8 48 85 C0 74 ?? 8B 54 [2] 48 8B C8 E8 [2] FF FF 48 8B CB FF [3] 00 00 }\n\t\t$c_rc4 =
{ FE C2 4C 8D 44 [2] 88 94 24 [2] 00 00 4D 8D 52 01 0F B6 C2 4C 03 C0 45 02 08 44 88 8C 24 [2] 00 00
41 0F B6 10 41 0F B6 C1 0F B6 4C [2] 41 88 08 0F B6 84 24 [2] 00 00 88 54 04 ??
44 0F B6 8C 24 [2] 00 00 }\n\t\t$c_exec = { 48 8B 4C 24 ?? 4C 8D 4C 24 ?? 45 33 C0 48 8B D? 4C 8B 11
41 FF 92 90 01 00 00 48 8B CB FF }\n\t\t$clr_clsid_metahost = { 8D 18 80 92 8E 0E 67 48 B3 0C 7F A8
38 84 E8 DE }\n\t\t$clr_iid_metahost = { 9E DB 32 D3 B3 B9 25 41
82 07 A1 48 84 F5 32 16 }\n\t\t$s1 = \"v4.0.30319\" wide\n\t\t$s2 = \"dat\" wide fullword\n\t\t$s3 =
\".tmp\" wide fullword\n\t\t$s4 = \"MapViewOfFileEx\"\n\t\t$s5 = \"CloseHandle\"\n\t\t$s6 = \"OLEAUT32\"
ascii wide\n\tcondition:\n\t\tuint16(0) == 0x5A4D and uint32(uint32(0x3C)) == 0x00004550 and any of
($deccall*) and any of ($c_*) and all of ($clr_*) and 4 of ($s*)\n}",
    "type": "signature"
  }
```


V4 YARA Endpoint

API Documentation

Purpose

This API endpoint lets you fetch YARA signature objects in bulk. This step provides you with complete rule details and associated attributes.

Key Parameters

- `limit` (optional): The maximum number of signatures to return (default 50, max 1000).
- `offset` (optional): Used for pagination, specifies the starting index for retrieval.

- `sort_by` (optional): Specifies the attribute to sort the signatures by (default created, supported values: `rule_name`, `modified`, `created`, `last_updated`).
- `sort_order` (optional): Specifies the sort order (`asc` or `desc`, default `desc`).
- `malware_ids` (optional): Comma-separated list of malware IDs to filter YARA signatures associated with.
- `malware_roles` (optional): Comma-separated list of malware roles to filter YARA signatures associated with.
- `start_created_epoch` (optional): Unix epoch timestamp for the start date to filter YARA signatures created after.
- `end_created_epoch` (optional): Unix epoch timestamp for the end date to filter YARA signatures created before.
- `modified_start_epoch` (optional): Unix epoch timestamp for the start date to filter YARA signatures modified after.
- `modified_end_epoch` (optional): Unix epoch timestamp for the end date to filter YARA signatures modified before.

Filters

- You can filter YARA signatures by malware IDs, roles, and creation/modification date ranges.
- Multiple filters can be combined for more granular results.

Response

- The response is a JSON object with the following properties:
  - `signatures`: An array of YARA signature objects, each containing details like rule name, meta-data, strings, and condition.
  - `total_count`: The total number of YARA signatures matching the query.
  - `next` (optional): A URL for retrieving the next page of results (when `enable_next_pagination` is used in the GET request).

Example Usage

- Fetch the 50 most recently created YARA signatures:

```
GET /v4/yara
```

- Fetch 1000 YARA signatures that are associated with malware IDs (using `malware--0c33c58c-d886-5d94-b777-4b91214394c6` and `malware--186de106-9460-56fe-bec1-fef201d56f45` in the example):

```
GET /v4/yara?malware_ids=malware--0c33c58c-d886-5d94-b777-4b91214394c6,
malware--186de106-9460-56fe-bec1-fef201d56f45&limit=1000
```

- Fetch YARA signatures created between January 1st and December 31st, 2023, sorted by modified date in descending order:

```
GET /v4/yara?start_created_epoch=1672531200&end_created_epoch=1708809600&sort_by=modified&sort_or
der=desc
```

- Fetch YARA signatures by role(s), sorted by modified date in descending order:

```
GET /v4/yara?malware_roles=Backdoor&sort_by=modified&sort_order=desc
```

- Get the next page of YARA signatures after the first 250:

```
GET v4/yara?limit=250&enable_next_pagination=true
GET /v4/yara?next=YOUR_NEXT_TOKEN
```

Additional Notes

- **Epoch Values**: The example usage provides epoch values based on today's date. Adjust these values based on your specific needs.
- Pagination is recommended for large result sets.

API Documentation for POST v4/yara

Purpose

Mandiant

This API endpoint allows you to fetch YARA signature objects in bulk, providing you with complete rule details and associated attributes.

Key Parameters

- `limit` (optional): The maximum number of signatures to return (default 50, max 1000).
- `offset` (optional): Used for pagination, specifies the starting index for retrieval.
- `enable_next_pagination` (optional): Enables the `next` property in the response for further pagination.
- `sort_by` (optional): Specifies the attribute to sort the signatures by (default `created`, supported values: `rule_name`, `modified`, `created`, `last_updated`).
- `sort_order` (optional): Specifies the sort order (`asc` or `desc`, default `desc`).
- `malware_ids` (optional): List of malware IDs to filter YARA signatures associated with.
- `malware_roles` (optional): List of malware roles to filter YARA signatures associated with.
- `start_created_epoch` (optional): Unix epoch timestamp for the start date to filter YARA signatures created after.
- `end_created_epoch` (optional): Unix epoch timestamp for the end date to filter YARA signatures created before.
- `modified_start_epoch` (optional): Unix epoch timestamp for the start date to filter YARA signatures modified after.
- `modified_end_epoch` (optional): Unix epoch timestamp for the end date to filter YARA signatures modified before.

Filters

- You can filter YARA signatures by malware IDs, roles, and creation/modification date ranges.
- Multiple filters can be combined for more granular results.

Response

- The response is a JSON object with the following properties:
  - `signatures`: An array of YARA signature objects, each containing details like rule name, meta-data, strings, and condition.
  - `total_count`: The total number of YARA signatures matching the query.
  - `next` (optional): A URL for retrieving the next page of results (when `enable_next_pagination` is true).

Example Usage

- Fetch the 50 most recently-created YARA signatures:

```
POST /v4/yara
```

- Fetch 100 YARA signatures associated with malware ID "malware--186de106-9460-56fe-bec1-fef201d56f45"

```
POST /v4/yara
{
  "malware_ids": ["malware--186de106-9460-56fe-bec1-fef201d56f45"],
  "limit": 100
}
```

- Fetch YARA signatures created between January 1 and December 31, 2023, sorted by modified date in descending order:

```
POST /v4/yara
{
  "start_created_epoch": 1672531200,
  "end_created_epoch": 1708809600,
  "sort_by": "modified",
  "sort_order": "desc"
}
```

- Get the next page of YARA signatures after the first 250:

```
GET /v4/yara?next=<URL from previous response>
```

Additional Notes

Pagination is supported for large result sets.

API Documentation for POST /v4/yara/download

Purpose

This API endpoint allows you to download YARA rules associated with a set of malware IDs in a ZIP file format.

Key Features

- Download YARA rules for multiple malware IDs at once.
- Filter downloaded rules by creation or modification date range.
- Receive a ZIP file containing valid YARA rules for each requested malware.

Key Parameters

- `malware_ids` (required): An array of malware IDs to download YARA rules for.
- `start_created_epoch` (optional): Unix timestamp for the start date to filter YARA rules created after.
- `end_created_epoch` (optional): Unix timestamp for the end date to filter YARA rules created before.
- `modified_start_epoch` (optional): Unix timestamp for the start date to filter YARA rules modified after.
- `modified_end_epoch` (optional): Unix timestamp for the end date to filter YARA rules modified before.
- `limit` (optional): Maximum number of YARA rules to download per malware (default 1000).
- `offset` (optional): Used for pagination within individual malware YARA rule sets.

Response

- Status code:
  - `200: Success` : ZIP file containing YARA rules for the requested malware.
  - `204: No Content` : No YARA rules found for the requested malware IDs or filters.
  - `400: Bad Request` : Invalid request body format or missing required parameters.
  - `404: Not Found` : Malware ID(s) not found.
  - `403: Forbidden` : User not authorized to access YARA download functionality.
- Response body (application/zip): A ZIP file containing individual YARA rule files named `yara-[alphanumeric].yar`. Each file contains the YARA rule data for a single malware.

Validation

- The ZIP file is validated to ensure it has valid structure and contains files with the expected naming format.
- Each individual YARA rule file is validated to ensure it has non-empty content and adheres to basic YARA syntax.

Additional Notes

- The limit and offset parameters apply to individual malware YARA rule sets within the ZIP file. They do not limit the total number of malware IDs you can request in a single download.

Example Usage of POST /v4/yara/download

Scenario: Download YARA rules for two malware IDs, filtering by creation date.

1. Request:

```
POST /v4/yara/download
Accept: application/zip
{
  "malware_ids": [
    "Malware-123",
    "Malware-456"
  ],
  "start_created_epoch": 1672531200, // January 1st, 2023
  "end_created_epoch": 1708809600 // December 31st, 2023
}
```

2.
3. ZIP file structure:
   The ZIP file contains two files:
   - `yara-Malware-123.yar` - Contains YARA rules for malware ID "Malware-123" created between January 1st and December 31st, 2023.
   - `yara-Malware-456.yar` - Contains YARA rules for malware ID "Malware-456" created between January 1st and December 31st, 2023.

V4 Malware ID YARA Endpoint

API Documentation for GET/POST /v4/malware/:id/yara

Purpose

This API endpoint allows you to retrieve YARA rule data associated with a specific malware identified by its ID.

Key Features

- Fetch YARA rules for a specific malware.
- Filter YARA rules by creation and modification date ranges.
- Retrieve only the most recent YARA rules (top N).

Key Parameters

- `malware_id` (required): The unique identifier of the malware you want to retrieve YARA rules for.
- `limit` (optional): The maximum number of YARA rules to return (default 10, max 100).
- `offset` (optional): Used for pagination, specifies the starting index for retrieval.
- `start_created_epoch` (optional): Unix epoch timestamp for the start date to filter YARA rules created after.
- `end_created_epoch` (optional): Unix epoch timestamp for the end date to filter YARA rules created before.
- `modified_start_epoch` (optional): Unix epoch timestamp for the start date to filter YARA rules modified after.
- `modified_end_epoch` (optional): Unix epoch timestamp for the end date to filter YARA rules modified before.

Response

- The response is a JSON object with the following properties:
  - `signatures` (array): An array of YARA signature objects associated with the malware, each containing details like rule name, meta-data, strings, and condition.
  - `total_count` (integer): The total number of YARA rules associated with the malware matching the query.
  - `next` (string, optional): A URL for retrieving the next page of results (when `enable_next_pagination` is used in the GET request).

Example Usage

- Retrieve the most recent 10 YARA rules associated with malware ID "ROYALLOCKER.BLACKSUIT":

```
GET /v4/malware/ROYALLOCKER.BLACKSUIT/yara?limit=10
```

- Fetch YARA rules associated with "ROYALLOCKER.BLACKSUIT", created between January 1st and December 31st, 2023:

```
GET /v4/malware/ROYALLOCKER.BLACKSUIT/yara?start_created_epoch=1672531200&end_created_epoch=1708809600
```

- Get the next page of YARA rules after the first 250:

```
GET /v4/yara?next=<URL_FROM_PREVIOUS_RESPONSE>
```

- Retrieve the most recent 10 YARA rules associated with malware ID "ROYALLOCKER.BLACKSUIT":

```
POST /v4/malware/ROYALLOCKER.BLACKSUIT/yara?limit=10
```

Additional Notes

- Pagination is supported for large result sets.
- POST `/v4/malware/:id/yara` takes the same parameters as GET