Инструкция по установке ПО «CTT Downloader»

Дистрибутивы

Для получения дистрибутива, ключа доступа и ссылки на поток данных, свяжитесь с нами: **info@cyberthreattech.ru**

ПО «CTT Downloader» предназначено для получения индикаторов из источников данных, в частном случае таким источником данных может являться облако ООО Технологии киберугроз (RST Cloud), и их загрузки, в том числе автоматической в различные базы данных, например, SIEM и Key-Value базы данных.

Системные требования

1. Свободное место на диске: 400 МБ.
2. ОС: Windows 10 (Home, Pro), Linux (Debian 9, 10).
3. Прямой доступ в интернет, или доступ через Proxy.

Порядок установки

1. Распаковать архив с ПО в требуемую директорию.
2. Отредактировать ./conf/config.yml (описание ниже).
    a. Указать пути в секции dirs
    b. Указать dirs.target – целевая директория для сохранения файла с индикаторами
    c. Указать dirs.tmp –временная директория для распаковки архива с индикаторами
3. Отредактировать ./conf/logging.yml (описание ниже).
    a. Минимальная настройка: Полный путь до лог-файла
4. Отредактировать ./conf/filters.yml (описание ниже).
    a. Минимальная настройка: Убрать комментарий (#) с тех полей, которые необходимо загружать в SIEM.
5. Дать права на исполнение cttdownloader (nix), или cttdownloader.exe (win)
   `chmod +x cttdownloader`

Запуск загрузчика

1. Запуск ПО для загрузки фида осуществляется командой:
   `./cttdownloader -f csv -t <ip|domain|url|hash>`

2. Для внешнего мониторинга (например, с использованием Zabbix, monitd) рекомендуется контролировать появления cttdownloader.alert в директории, указанной в конфигурационном файле. Данный файл создается в случае, если работа ПО завершилась с ошибкой.
3. Добавить запуск ПО в cron. В облаке фиды обновляются 1 раз в сутки в интервале с 04:00 - 04:30 по Мск. Т.о. загрузку рекомендуется планировать на 05:00 по Мск.
4. При добавлении ПО загрузки в cron необходимо задать каталог ПО в качестве рабочей директории, либо явно определить параметры `-c, -e, -l`
5. После запуска, в директории target (см. config.yml) появится файл, подготовленный для загрузки в приложение для использования полученных данных, например, в СЗИ.

Параметры запуска

```
-c, --config                            Путь до config.yml
                                        По умолчанию: ./conf/config.yml

-e, --filters                           Путь до filters.yml.
                                        По умолчанию: ./conf/filters.yml

-l, --log                               Путь до logging.yml.
                                        По умолчанию: ./conf/logging.yml

-f, --convformat    [обязательный]      Формат для конвертирования фида.
                                        По умолчанию: none
                                        none – не конвертировать, только скачать.
                                        csv – Конвертировать в CSV-формат
                                        mpsiem – Конвертировать в формат MP SIEM
                                        qradar – Конвертировать в формат IBM QRadar
                                        arcsight - Конвертировать в формат Micro Focus ArcSight
                                        redis - Конвертировать в формат Redis
                                        memcached - Конвертировать в формат Memcached

-t, --feedtype      [обязательный]      Тип фида.
                                        ip - IP-адреса
                                        domain – Домены
                                        url – URL
                                        hash - Хэши

--upload / --not-upload                 Флаг. Попытаться установит в SIEM (только для mpsiem и
                                        qradar).
                                        По умолчанию: --upload

--cleartmp / --not-cleartmp             Флаг. Очищать временную директорию после
                                        успешной конвертации.
                                        По умолчанию: --cleartmp

--cday <число>                          Принудительно выставить дату, за которую надо искать
                                        файл фида.
                                        Unix timestamp (int 32)

--help                                  Показать справку
```

Описание конфигурационных файлов

config.yml

```
connection:
  proxy:          # Удалить, если proxy не используется
    type: 'https' # http/https
    url: 'socks5://123.123.123.123:8888' # format <http/socks5>://<user:pass>@<ip/fqdn>:<port>

  cttcloud:
    baseurl: '' # Ваша ссылка на поток данных
    apikey: '' # Ваш ключ для подключения
    contimeout: 10   # Можно повысить, если большой пинг до сервера
    readtimeout: 20 # Можно повысить, если медленное соединение
    retry: 2        # Кол-во попыток переподключения
    delete_gz: true
    feeds:
      filetype: 'json'

dirs:
  target: './target' # Целевая директория для сконвертированного файла
  tmp: './tmp '      # Временная директория для скаченного файла. Очищается после конвертации
  alert: './'        # Директория для cttdownloader.alert
  state: './ '       # Директория для cttdownloader.state
```

filters.yml

```
export:
  fields:                   # Список экспортируемых из фида полей.
    ip:
      #0: '_last_changed' # для MP SIEM ( < R24)
      1: 'ip_v4'
      #2: 'ip_num'
      3: 'fseen'
      4: 'lseen'
      5: 'collect'
      8: 'tags_str'
      #9: 'tags_codes'
      10: 'asn_num'
      #11: 'asn_firstip_netv4'
      #12: 'asn_firstip_num'
      #13: 'asn_lastip_netv4'
      #14: 'asn_lastip_num'
      15: 'asn_cloud'
      16: 'asn_domains'
      17: 'asn_org'
      18: 'asn_isp'
      #19: 'geo_city'
      20: 'geo_country'
      #21: 'geo_region'
      22: 'related_domains'
      24: 'score_src'
      25: 'score_tags'
      26: 'score_frequency'
      27: 'score_total'
      28: 'fp_alarm'
      29: 'fp_descr'
      30: 'threat'
      31: 'cve'
      32: 'industry'
      33: 'src_report'
      34: 'id'
      35: 'title'
      36: 'description'
      37: 'ports'
    domain:
      #0: '_last_changed' # для MP SIEM ( < R24)
      1: 'domain'
      3: 'fseen'
      4: 'lseen'
      5: 'collect'
      8: 'tags_str'
      #9: 'tags_codes'
      10: 'resolved_ip_a'
      11: 'resolved_ip_alias'
      12: 'resolved_ip_cname'
      13: 'resolved_whois_created'
      14: 'resolved_whois_updated'
      #15: 'resolved_whois_expires'
      #16: 'resolved_whois_age'
      17: 'resolved_whois_registrar'
      18: 'resolved_whois_registrant'
      #19: 'resolved_whois_havedata'
      20: 'score_src'
      21: 'score_tags'
      22: 'score_frequency'
      23: 'score_total'
      24: 'fp_alarm'
      25: 'fp_descr'
```

```
    26: 'threat'
    27: 'cve'
    28: 'industry'
    29: 'src_report'
    30: 'id'
    31: 'title'
    32: 'description'
    33: 'ports'
  url:
    #0: '_last_changed' # для MP SIEM ( < R24)
    1: 'url'
    2: 'fseen'
    3: 'lseen'
    4: 'collect'
    5: 'tags_str'
    #6: 'tags_codes'
    7: 'score_src'
    8: 'score_tags'
    9: 'score_frequency'
    10: 'parsed_schema'
    11: 'parsed_domain'
    12: 'parsed_port'
    13: 'parsed_path'
    14: 'parsed_params'
    15: 'parsed_anchor'
    16: 'resolved_status'
    17: 'score_total'
    18: 'fp_alarm'
    19: 'fp_descr'
    20: 'threat'
    21: 'cve'
    22: 'industry'
    23: 'src_report'
    24: 'id'
    25: 'title'
    26: 'description'
  hash:
    #0: '_last_changed' # для MP SIEM ( < R24)
    1: 'fseen'
    2: 'lseen'
    3: 'collect'
    4: 'md5'
    5: 'sha1'
    6: 'sha256'
    7: 'filename'
    8: 'tags_str'
    #9: 'tags_codes'
    10: 'score_src'
    11: 'score_tags'
    12: 'score_frequency'
    13: 'score_total'
    14: 'fp_alarm'
    15: 'fp_descr'
    16: 'threat'
    17: 'cve'
    18: 'industry'
    19: 'src_report'
    20: 'id'
    21: 'title'
    22: 'description'
falsealarm:      # Экспорт индикаторов с определенным уровнем ложных срабатываний.
  0: 'true'
  1: 'false'
  2: 'possible'
```

```
  tags:            # Оставлять только индикаторы с любым из указанных тегов
    - c2
    - malware
  score:
    threshold: 40  # Отбросить индикаторы со score_total меньше указанного
    highrisk: 40   # Применимо для QRadar. Создавать отдельный Set с индикаторами,
                    чей score_total выше указанного.
  ignorefilters:
    keeptags:      # Игнорировать ограничения на score и оставлять индикаторы с любым,
                     указанным, тегом
      - tor_node
      - tor_exit
```

logging.yml

```
version: 1
disable_existing_loggers: True
formatters:
  simple:
    format: "%(asctime)s - %(name)s - %(levelname)s - %(message)s"
handlers:
  console:
    class: logging.StreamHandler
    level: DEBUG
    formatter: simple
    stream: ext://sys.stdout
  file_handler:
    class: logging.handlers.RotatingFileHandler
    level: DEBUG
    formatter: simple
    filename: ./var/log/cttdownloader.log #Установить путь до файла лога. Убедиться, что есть
права на запись в директорию
    maxBytes: 10485760 # 10MB
    backupCount: 10
    encoding: utf8
loggers:
  cttdownloader:
    level: INFO      # Изменить уровень журналирования, если необходимо
    handlers: [console, file_handler]
    propagate: no
root:
  level: INFO
  handlers: [console, file_handler]
```