Windows.Carving.SystemBC

This artifact extracts SystemBC RAT configuration from a byte stream, process or file on disk.

The User can define bytes, file glob, process name or pid regex as a target. The artifact firstly discovers configuration and extracts bytes, before parsing with Velociraptor Binary Parser.

This content simply carves the configuration and does not unpack files on disk. That means pointing this artifact as a packed or obfuscated file may not obtain the expected results.

```
name: Windows.Carving.SystemBC
author: Matt Green - @mgreen27
description: |
  This artifact extracts SystemBC RAT configuration from a byte stream,
  process or file on disk.

  The User can define bytes, file glob, process name or pid regex as a target.
  The artifact firstly discovers configuration and extracts bytes,
  before parsing with Velociraptor Binary Parser.

  This content simply carves the configuration and does not unpack files on
  disk. That means pointing this artifact as a packed or obfuscated file may not
  obtain the expected results.

reference:
  - https://malpedia.caad.fkie.fraunhofer.de/details/win.systembc


parameters:
  - name: TargetBytes
    default:
  - name: TargetFileGlob
    default:
  - name: PidRegex
    default: .
    type: regex
  - name: ProcessRegex
    default: .
    type: regex
  - name: FindConfig
    type: hidden
    description: Final Yara option and the default if no other options provided.
    default: |
        rule SystemBC_Config
            {
                meta:
                    author = "Matt Green - @mgreen27"
                    description = "SystemBC configuration"

                strings:
                    $BEGINDATA = { 00 42 45 47 49 4e 44 41 54 41 00 } //BEGINDATA
                    $ = "HOST1:" ascii wide fullword
                    $ = "HOST2:" ascii wide fullword
                    $ = "PORT1:" ascii wide fullword
                    $ = "TOR:" ascii wide fullword
                    $ = "-WindowStyle Hidden -ep bypass -file" ascii wide

                condition:
                    $BEGINDATA and 3 of them
            }
sources:
  - precondition:
```

```
      SELECT OS From info() where OS = 'windows'

    query: |
      -- binary parse profile to extract SystemBC configuration.
      LET PROFILE = '''[
                [SystemBC, 0, [
                    ["__FindHost1",0, "String",{term: "HOST1:"}],
                    ["HOST1","x=>len(list=x.__FindHost1) + 6", "String",{term_hex:
"0000"}],
                    ["__FindHost2",0, "String",{term: "HOST2:"}],
                    ["HOST2","x=>len(list=x.__FindHost2) + 6", "String",{term_hex:
"0000"}],
                    ["__FindPort1",0, "String",{term: "PORT1:"}],
                    ["PORT1","x=>len(list=x.__FindPort1) + 6", "String",{term_hex:
"0000"}],
                    ["__FindTOR",0, "String",{term: "TOR:"}],
                    ["TOR","x=>len(list=x.__FindTOR) + 4", "String",{term_hex:
"0000"}],
                    ["__FindUserAgent",0, "String",{term: "\r\nUser-Agent: "}],
                    ["User-Agent","x=>len(list=x.__FindUserAgent) + 14", "String",
{term: "\r\n"}],
                ]
            ]]'''


      -- Bytes usecase: scan DataBytes for config
      LET ByteConfiguration = SELECT
            Rule,
            len(list=TargetBytes) as Size,
            hash(path=TargetBytes,accessor='data') as Hash,
            String.Offset as HitOffset,
            read_file(accessor="data",filename=TargetBytes, offset=String.Offset,
length=1000) as _RawConfig
        FROM yara(
                files=TargetBytes,
                accessor='data',
                rules=FindConfig,
                number=99,
                context=1000
            )
        GROUP BY _RawConfig

      -- Glob usecase: find target files
      LET TargetFiles = SELECT OSPath,Size
        FROM glob(globs=TargetFileGlob) WHERE NOT IsDir

      -- Glob usecase: Extract config from files in scope
      LET FileConfiguration = SELECT * FROM foreach(row=TargetFiles,
            query={
                SELECT
                    Rule,
                    OSPath, Size,
```

```
                    hash(path=OSPath) as Hash,
                    String.Offset as HitOffset,
                    read_file(filename=OSPath, offset=String.Offset, length=1000) as
_RawConfig
                FROM yara(
                        files=OSPath,
                        rules=FindConfig,
                        number=99,
                        context=1000
                    )
                GROUP BY OSPath,_RawConfig
            })

      -- find velociraptor process
      LET me <= SELECT * FROM if(condition= NOT ( TargetFileGlob OR TargetBytes ),
                    then = { SELECT Pid FROM pslist(pid=getpid()) })

      -- find all processes and add filters
      LET processes = SELECT Name as ProcessName, Exe, CommandLine, Pid
        FROM pslist()
        WHERE
            Name =~ ProcessRegex
            AND format(format="%d", args=Pid) =~ PidRegex
            AND NOT Pid in me.Pid

      -- scan processes in scope with our rule, limit 1 hit and extract context to
parse
      LET ProcessConfiguration = SELECT * FROM foreach(
        row=processes,
        query={
            SELECT
                Rule,
                Pid, ProcessName, CommandLine,
                String.Offset as HitOffset,
                read_file(accessor="process", filename=format(format="/%d",
args=Pid), offset=String.Offset, length=1000) as _RawConfig
            FROM yara(
                    files=format(format="/%d", args=Pid),
                    accessor='process',
                    rules=FindConfig,
                    number=99,
                    context=1000
                )
            GROUP BY Pid, ProcessName, CommandLine,_RawConfig
          })



      -- generate results remove any FPs
      SELECT *,
        parse_binary(accessor="data", filename=_RawConfig, profile=PROFILE,
struct='SystemBC') AS SystemBC,
```

```
  _RawConfig
FROM if(condition=TargetBytes,
      then=ByteConfiguration,
      else= if(condition=TargetFileGlob,
          then= FileConfiguration,
          else= ProcessConfiguration))
WHERE SystemBC.HOST1 OR SystemBC.HOST2 OR SystemBC.TOR
```