```
{
    "version": "3.0",
    "name": "ServiceNow_Security_Events",
    "comment": "Create an incident by a DNS security events",
    "type": "REST_EVENT",
    "event_type": [
        "RPZ",
        "TUNNEL"
    ],
    "action_type": "Incidents",
    "content_type": "application/json",
    "vendor_identifier": "ServiceNow",
    "quoting": "XMLA",
    "instance_variables": [
        {
            "name": "Severity",
            "type": "INT",
            "value": "3"
        }
    ],
    "steps": [
        {
            "name": "assignTimeValue",
            "operation": "NOP",
            "body_list": [
                "${XC:COPY:{L:ServiceNowAddDate}:
{UT:TIME}}${XC:FORMAT:TRUNCATE:{L:ServiceNowAddDate}:{10t}}"
            ]
        },
        {
            "name": "check for IPv6",
            "operation": "CONDITION",
            "condition": {
                "statements": [
                    {
                        "left": "${E::source_ip}",
                        "op": "=~",
                        "right": ":"
                    }
                ],
                "condition_type": "AND",
                "next": "Get IPv6Fixed _ref"
```

```
            }
        },
        {
            "name": "Debug#14",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "name": "Get IPv4Fixed _ref",
            "operation": "GET",
            "transport": {
                "path": "fixedaddress?ipv4addr=${E:U:source_ip}
&network_view=default&_return_fields=extattrs"
            },
            "wapi": "v2.7"
        },
        {
            "name": "Debug#15",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "operation": "CONDITION",
            "name": "wapi_response_getIPv4Fix_ref",
            "condition": {
                "condition_type": "AND",
                "statements": [
                    {
                        "left": "${P:A:PARSE[0]{_ref}}",
                        "op": "!=",
                        "right": ""
                    }
                ],
                "next": "Get_Objref"
            }
        },
        {
            "name": "Debug#16",
```

```
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "name": "Get HostIPv4 _ref",
            "operation": "GET",
            "transport": {
                "path": "record:host?ipv4addr=${E:U:source_ip}
&network_view=default&_return_fields=extattrs"
            },
            "wapi": "v2.7"
        },
        {
            "name": "Debug#17",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "operation": "CONDITION",
            "name": "wapi_response_getIPv4Host_ref",
            "condition": {
                "condition_type": "AND",
                "statements": [
                    {
                        "left": "${P:A:PARSE[0]{_ref}}",
                        "op": "!=",
                        "right": ""
                    }
                ],
                "next": "Get_Objref",
                "else_stop": true
            }
        },
        {
            "name": "Debug#19",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
```

```
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "name": "Get IPv6Fixed _ref",
            "operation": "GET",
            "transport": {
                "path": "ipv6fixedaddress?ipv6addr=${E:U:source_ip}
&network_view=default&_return_fields=extattrs"
            },
            "wapi": "v2.7"
        },
        {
            "name": "Debug#20",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "operation": "CONDITION",
            "name": "wapi_response_getIPv6Fix_ref",
            "condition": {
                "condition_type": "AND",
                "statements": [
                    {
                        "left": "${P:A:PARSE[0]{_ref}}",
                        "op": "!=",
                        "right": ""
                    }
                ],
                "next": "Get_Objref"
            }
        },
        {
            "name": "Debug#21",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "name": "Get HostIPv6 _ref",
```

```
            "operation": "GET",
            "transport": {
                "path": "record:host?ipv6addr=${E:U:source_ip}
&network_view=default&_return_fields=extattrs"
            },
            "wapi": "v2.7"
        },
        {
            "name": "Debug#22",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "operation": "CONDITION",
            "name": "wapi_response_getIPv6Host_ref",
            "condition": {
                "condition_type": "AND",
                "statements": [
                    {
                        "left": "${P:A:PARSE[0]{_ref}}",
                        "op": "!=",
                        "right": ""
                    }
                ],
                "next": "Get_Objref"
            }
        },
        {
            "name": "Debug#23",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
        },
        {
            "name": "Get_Objref",
            "operation": "CONDITION",
            "condition": {
                "condition_type": "AND",
                "statements": [
```

```
                {
                    "left": "${P:A:PARSE[0]{_ref}}",
                    "op": "!=",
                    "right": ""
                }
            ],
            "eval": "${XC:COPY:{L:Obj_ref}:{P:PARSE[0]{_ref}}}"
        }
    },
    {
        "name": "Debug#24",
        "operation": "NOP",
        "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
    },
    {
        "name": "Assign location variable",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "OR",
            "statements": [
                {
                    "left": "${P:A:PARSE[0]{extattrs}{ServiceNow_Location}
{value}}",
                    "op": "==",
                    "right": ""
                }
            ],
            "eval": "${XC:ASSIGN:{L:Location}:{S:Unknown}}",
            "else_eval": "${XC:COPY:{L:Location}:{P:PARSE[0]{extattrs}
{ServiceNow_Location}{value}}}"
        }
    },
    {
        "name": "jump if no Obj_ref",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "OR",
            "statements": [
                {
                    "left": "${L:A:Obj_ref}",
```

```
                    "op": "==",
                    "right": ""
                }
            ],
            "next": "check rpz or tunnel to assign query name"
        }
    },
    {
        "name": "Debug#25",
        "operation": "NOP",
        "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
    },
    {
        "name": "stop if no extattrs",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "OR",
            "statements": [
                {
                    "left": "${P:A:PARSE[0]{extattrs}
{ServiceNow_Add_Incident}{value}}",
                    "op": "==",
                    "right": ""
                }
            ],
            "stop": true
        }
    },
    {
        "name": "assignRecordValues",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "OR",
            "statements": [
                {
                    "left": "${P:A:PARSE[0]{extattrs}
{ServiceNow_LastIncidentSentAt}{value}}",
                    "op": "==",
                    "right": ""
                }
```

```
            ],
            "eval": "${XC:ASSIGN:{L:ServiceNowAddDateRecorded}:
{S:NONE}}",
            "else_eval": "${XC:COPY:{L:ServiceNowAddDateRecorded}:
{P:PARSE[0]{extattrs}{ServiceNow_LastIncidentSentAt}
{value}}}${XC:FORMAT:TRUNCATE:{L:ServiceNowAddDateRecorded}:
{10t}}"
         }
      },
      {
         "name": "check If Scan Happened today",
         "operation": "CONDITION",
         "condition": {
            "condition_type": "OR",
            "statements": [
               {
                  "left": "${P:A:PARSE[0]{extattrs}
{ServiceNow_Add_Incident}{value}}",
                  "op": "==",
                  "right": "false"
               },
               {
                  "left": "${L:A:ServiceNowAddDateRecorded}",
                  "op": "==",
                  "right": "${L:A:ServiceNowAddDate}"
               }
            ],
            "stop": true
         }
      },
      {
         "name": "check rpz or tunnel to assign query name",
         "operation": "CONDITION",
         "condition": {
            "statements": [
               {
                  "left": "${E::event_type}",
                  "op": "==",
                  "right": "RPZ"
               }
            ],
            "condition_type": "AND",
```

```
            "eval": "${XC:COPY:{L:query_name}:{E:query_name}}",
            "else_eval": "${XC:COPY:{L:query_name}:{E:domain_name}}"
        }
    },
    {
        "name": "Debug#4",
        "operation": "NOP",
        "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
    },
    {
        "name": "set threatActionTaken threatHandled",
        "operation": "CONDITION",
        "condition": {
            "statements": [
                {
                    "left": "${E::rpz_policy}",
                    "op": "==",
                    "right": "PASSTHRU"
                }
            ],
            "condition_type": "AND",
            "eval": "${XC:ASSIGN:{L:threatActionTaken}:{S:Recorded}}",
            "else_eval": "${XC:ASSIGN:{L:threatActionTaken}:{S:Blocked}}"
        }
    },
    {
        "name": "Create an incident",
        "operation": "POST",
        "parse": "JSON",
        "transport": {
            "path": "/api/now/v2/table/incident"
        },
        "body_list": [
            "{",
            "\"category\":\"Network Security\",",
            "\"subcategory\":\"DNS ${E:A:event_type}\",",
            "\"description\":\"Client ${E:A:source_ip} accessed restricted
domain ${L:A:query_name} and this was ${L:A:threatActionTaken} by
Infoblox appliance with IP:${E:A:member_ip}\",",
            "\"short_description\":\"Client ${E:A:source_ip} accessed
```

```
restricted domain ${L:A:query_name} and this was
${L:A:threatActionTaken} by Infoblox appliance with IP:
${E:A:member_ip}\",",
                "\"severity\":\"${I:A:Severity}\",",
                "\"u_Location\":\"${L:A:Location}\",",
                "\"sys_created_by\":\"NIOS Outbound API\"",
                "}"
            ]
        },
        {
            "name": "Debug",
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{R:}}${XC:DEBUG:{RH:}}${XC:DEBUG:{UT:}}"
        },
        {
            "name": "Incident creation error check",
            "operation": "CONDITION",
            "condition": {
                "condition_type": "AND",
                "statements": [
                    {
                        "left": "${R:A:RC}",
                        "op": "!=",
                        "right": "201"
                    }
                ],
                "error": true
            }
        },
        {
            "name": "Get the incident",
            "operation": "GET",
            "parse": "JSON",
            "transport": {
                "path": "/api/now/v2/table/incident?number=
${P:U:result{number}}"
            }
        },
        {
            "name": "Debug Incident",
```

```
            "operation": "NOP",
            "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{R:}}${XC:DEBUG:{RH:}}${XC:DEBUG:{UT:}}"
        },
        {
            "name": "set time incedent was created to a variable",
            "operation": "NOP",
            "body_list": [
                "${XC:COPY:{L:TimeIncidentCreated}:{P:result[0]
{sys_created_on}}}"
            ]
        },
        {
            "name": "jump if no Obj_ref2",
            "operation": "CONDITION",
            "condition": {
                "condition_type": "OR",
                "statements": [
                    {
                        "left": "${L:A:Obj_ref}",
                        "op": "==",
                        "right": ""
                    }
                ],
                "next": "done"
            }
        },
        {
            "name": "Update timestamp and system ID",
            "operation": "PUT",
            "transport": {
                "path": "${L:A:Obj_ref}"
            },
            "wapi": "v2.7",
            "wapi_quoting": "JSON",
            "body_list": [
                "{\"extattrs+\":{\"ServiceNow_LastIncidentSentAt\": { \"value\":
\"${L:A:TimeIncidentCreated}\"},\"ServiceNow_Event_ID\": { \"value\":
\"${P:A:result[0]{number}}\"}}}"
            ]
        },
```

```
        {
            "name": "done",
            "operation": "NOP",
            "body": ""
        }
    ]
}
```