```
{
    "version": "4.0",
    "name": "ServiceNow_Security_Incident_Events",
    "comment": "Create an incident by a DNS security events",
    "type": "REST_EVENT",
    "event_type": [
        "RPZ",
        "TUNNEL",
        "ADP"
    ],
    "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": "${L:A:Obj_ref}",
                        "op": "==",
                        "right": ""
                    }
                ],
                "next": "check if ADP event"
            }
        },
        {
            "name": "Debug#25",
            "operation": "NOP",
            "body": "${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 if ADP event",
            "operation": "CONDITION",
            "condition": {
                "statements": [
                    {
                        "left": "${E::event_type}",
                        "op": "==",
                        "right": "ADP"
                    }
                ],
                "condition_type": "AND",
                "eval": "${XC:COPY:{L:query_name}:{E:query_fqdn}}",
                "else_next":"done"
            }
        },

        {
            "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 for ADP",
            "operation": "POST",
            "parse": "JSON",
            "transport": {
                "path": "/api/now/v2/table/incident"
            },
            "body_list": [
                "{",
                    "\"category\":\"Network Security\",",
                    "\"subcategory\":\"DNS ${E:A:event_type}\",",
                    "\"description\":\"Event Information:\\n
Infoblox appliance: ${E:A:member_name}, ${E:A:member_ip} \\nQuery
FQDN: ${E:A:query_fqdn},\\n Timestamp: ${E:A:timestamp}\\n\\n Rule
Name: ${E:A:rule_name}, Rule Id: ${E:A:rule_sid}, Rule Category: $
{E:A:rule_category}, Rule Action: ${E:A:rule_action}, Rule Severity:
```

```
${E:A:rule_severity}\",",
                    "\"short_description\":\"Client ${E:A:source_ip}
has an ADP event with action ${E:A:rule_action} and ruleid of $
{E:A:rule_sid} applied by Infoblox appliance ${E:A:member_name}\",",
                    "\"severity\":\"${I:A:Severity}\",",
                    "\"u_Location\":\"${L:A:Location}\",",
                    "\"contact_type\":\"Network Monitoring\",",
                    "\"sys_created_by\":\"NIOS Outbound API\",",
                    "\"work_notes\":\"event type: ${E:A:event_type},
Rule Action: ${E:A:rule_action}\"",
                    "}"
            ]
        },
        {
            "name": "Debug",
            "operation": "NOP",
            "body": "${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": "${R:A:RC}",
                        "op": "!=",
                        "right": "201"
                    }
                ],
                "error": true
            }
        },
        {
            "name": "Debug Incident",
            "operation": "NOP",
            "body": "${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": "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:ServiceNowAddDate}\"}}}"
            ]
        },
        {
            "name": "done",
            "operation": "NOP",
            "body": ""
        }
    ]
}
```