```json
{
    "name": "DXL_Host_Event",
    "version": "3.0",
    "type": "DXL_EVENT",
    "event_type": [
        "HOST_ADDRESS_IPV4",
        "HOST_ADDRESS_IPV6"
    ],
    "vendor_identifier": "McAfee",
    "quoting": "ASIS",
    "instance_variables": [
        {
            "name": "DXL_MessageFormat",
            "type": "STRING"
        },
        {
            "name": "OPERATION_TYPES",
            "type": "STRING",
            "value": "insert/modify/delete"
        }
    ],
    "steps": [
        {
            "name": "Debug#0",
            "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 time vars",
            "operation": "NOP",
            "body_list": [
                "${XC:COPY:{L:New_Time}:{E:timestamp}}${XC:FORMAT:TRUNCATE:{L:New_Time}:{16t}}"
            ]
        },
        {
            "name": "Set Old_Time",
            "operation": "CONDITION",
            "condition": {
                "condition_type": "OR",
```

```
      "statements": [
        {
          "left": "${E:A:values{extattrs}{DXL_LastEventSentAt}
{value}}",
          "op": "==",
          "right": ""
        }
      ],
      "eval": "${XC:ASSIGN:{L:Old_Time}:{S:}}",
      "else_eval": "${XC:COPY:{L:Old_Time}:{E:values{extattrs}
{DXL_LastEventSentAt}{value}}}${XC:FORMAT:TRUNCATE:
{L:Old_Time}:{16t}}"
    }
  },
  {
    "name": "Debug#1",
    "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 modified in the last second",
    "operation": "CONDITION",
    "condition": {
      "condition_type": "OR",
      "statements": [
        {
          "left": "${L:A:New_Time}",
          "op": "==",
          "right": "${L:A:Old_Time}"
        }
      ],
      "stop": true
    }
  },
  {
    "name": "Debug#2",
    "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:}}"
```

```json
        },
        {
            "name": "STOP if sync not requested",
            "operation": "CONDITION",
            "condition": {
                "condition_type": "OR",
                "statements": [
                    {
                        "left": "${E:A:values{extattrs}{DXL_Sync}{value}}",
                        "op": "==",
                        "right": ""
                    },
                    {
                        "left": "${E:A:values{extattrs}{DXL_Sync}{value}}",
                        "op": "==",
                        "right": "false"
                    }
                ],
                "stop": true
            }
        },
        {
            "name": "init_internal_data",
            "operation": "VARIABLEOP",
            "variable_ops": [
                {
                    "operation": "ASSIGN",
                    "type": "DICTIONARY",
                    "destination": "L:internal",
                    "keys": [
                        "analyzer_ipv4",
                        "analyzer_ipv6",
                        "source_ipv4",
                        "source_ipv6",
                        "target_ipv4",
                        "target_ipv6",
                        "severity"
                    ],
                    "values": [
                        "",
                        "",
                        "",
```

```json
                        "",
                        "",
                        "",
                        "7"
                    ]
                }
            ]
        },
        {
            "name": "check what operation types are allowed",
            "operation": "CONDITION",
            "condition": {
                "condition_type": "AND",
                "statements": [
                    {
                        "left": "${I::OPERATION_TYPES}",
                        "op": "!~",
                        "right": "((?i).*${E::operation_type}.*)"
                    }
                ],
                "next": "Fin"
            }
        },
        {
            "name": "is_analyzer_source_HOST_ipv4",
            "operation": "CONDITION",
            "condition": {
                "statements": [
                    {
                        "left": "${E::object_type}",
                        "op": "==",
                        "right": "HostAddress"
                    },
                    {
                        "left": "${E::values{ipv4addr}}",
                        "op": "!=",
                        "right": ""
                    }
                ],
                "condition_type": "AND",
                "eval": "${XC:COPY:{L:internal{analyzer_ipv4}}:
{E:member_ip}}${XC:COPY:{L:internal{source_ipv4}}:
```

```
{E:member_ip}}${XC:ASSIGN:{L:IPv}:{I:4}}",
          "else_eval": "${XC:COPY:{L:internal{analyzer_ipv6}}:
{E:member_ip}}${XC:COPY:{L:internal{source_ipv6}}:
{E:member_ip}}${XC:ASSIGN:{L:IPv}:{I:6}}"
        }
    },
    {
        "name": "is_target_ipv4",
        "operation": "CONDITION",
        "condition": {
          "statements": [
              {
                  "left": "${E::values{ipv4addr}}",
                  "op": "!=",
                  "right": ""
              }
          ],
          "condition_type": "AND",
          "eval": "${XC:COPY:{L:internal{target_ipv4}}:
{E:values{ipv4addr}}}",
          "else_eval": "${XC:COPY:{L:internal{target_ipv6}}:
{E:values{ipv6addr}}}"
        }
    },
    {
        "name": "is_severity_7",
        "operation": "CONDITION",
        "condition": {
          "statements": [
              {
                  "left": "1",
                  "op": "==",
                  "right": "1"
              }
          ],
          "condition_type": "AND",
          "eval": "${XC:ASSIGN:{L:internal{severity}}:{I:7}}"
        }
    },
    {
        "name": "set some host variables and DetectedUTC",
        "operation": "NOP",
```

```
        "body_list": [
          "${XC:COPY:{L:ruleName}:
{E:member_name}}${XC:FORMAT:TRUNCATE:{L:ruleName}:{-128f}}",
          "${XC:COPY:{L:threatName}:
{E:values{host}}}${XC:FORMAT:TRUNCATE:{L:threatName}:{-128f}}",
          "${XC:COPY:{L:DetectedUTC}:{E:timestamp}}",
          "${XC:ASSIGN:{L:Obj_ref}:{S:}}${XC:ASSIGN:{L:network_view}:
{S:default}}",
          "${XC:COPY:{L:Object_type}:{E:object_type}}",
          "${XC:COPY:{L:operation_type}:{E:operation_type}}"
        ]
      },
      {
        "name": "check GUID",
        "operation": "CONDITION",
        "condition": {
          "condition_type": "OR",
          "statements": [
            {
              "left": "${E:A:values{extattrs}{ePO_GUID}{value}}",
              "op": "==",
              "right": ""
            }
          ],
          "eval": "${XC:COPY:{L:GUID}:{UT:UUID}}${XC:ASSIGN:
{L:GUIDtype}:{S:generated}}",
          "else_eval": "${XC:COPY:{L:GUID}:{E:values{extattrs}
{ePO_GUID}{value}}}${XC:ASSIGN:{L:GUIDtype}:{S:local}}"
        }
      },
      {
        "name": "jump if have GUID or no WAPI credentials or is delete",
        "operation": "CONDITION",
        "condition": {
          "condition_type": "OR",
          "statements": [
            {
              "left": "${L:A:GUIDtype}",
              "op": "==",
              "right": "local"
            },
            {
```

```json
              "left": "${UT:A:WAPIUSERNAME}",
              "op": "==",
              "right": ""
            },
            {
              "left": "${E:A:operation_type}",
              "op": "==",
              "right": "DELETE"
            }
          ],
          "next": "Check if operation type was delete to avoid errors"
        }
      },
      {
        "name": "Check if operation type was delete to avoid errors",
        "operation": "CONDITION",
        "condition": {
          "condition_type": "AND",
          "statements": [
            {
              "left": "${E:A:operation_type}",
              "op": "==",
              "right": "DELETE"
            }
          ],
          "next": "check DXL_MessageFormat_Delete"
        }
      },
      {
        "name": "set up address",
        "operation": "CONDITION",
        "condition": {
          "condition_type": "AND",
          "statements": [
            {
              "left": "${E::values{ipv4addr}}",
              "op": "!=",
              "right": ""
            }
          ],
          "eval": "${XC:COPY:{L:IP}:{E:values{ipv4addr}}}",
          "else_eval": "${XC:COPY:{L:IP}:{E:values{ipv6addr}}}"
```

```json
      }
    },
    {
      "name": "Get User Data",
      "operation": "GET",
      "transport": {
        "path": "networkuser?user_status=ACTIVE&address=${L:A:IP}"
      },
      "wapi": "v2.6"
    },
    {
      "name": "check_user_response",
      "operation": "CONDITION",
      "condition": {
        "condition_type": "AND",
        "statements": [
          {
            "left": "${P:L:PARSE}",
            "op": "==",
            "right": "0"
          }
        ],
        "next": "check_username"
      }
    },
    {
      "name": "Pop User from the list",
      "operation": "VARIABLEOP",
      "variable_ops": [
        {
          "operation": "UNSHIFT",
          "type": "DICTIONARY",
          "destination": "L:user",
          "source": "P:PARSE"
        }
      ]
    },
    {
      "name": "check_username",
      "operation": "CONDITION",
      "condition": {
        "condition_type": "AND",
```

```
        "statements": [
          {
              "left": "${L::user{name}}",
              "op": "!=",
              "right": ""
          }
        ],
        "eval": "${XC:COPY:{L:username}:{L:user{name}}}${XC:COPY:
{L:domainname}:{L:user{domainname}}}",
        "else_eval": "${XC:ASSIGN:{L:username}:{S:.}}${XC:ASSIGN:
{L:domainname}:{S:.}}"
      }
    },
    {
      "name": "assign ipv4 or ipv6 ip to use for GET requests",
      "operation": "CONDITION",
      "condition": {
        "condition_type": "AND",
        "statements": [
          {
              "left": "${E::values{ipv4addr}}",
              "op": "!=",
              "right": ""
          }
        ],
        "eval": "${XC:COPY:{L:GetIP}:{E:values{ipv4addr}}}",
        "else_eval": "${XC:COPY:{L:GetIP}:{E:values{ipv6addr}}}"
      }
    },
    {
      "name": "check IPv6",
      "operation": "CONDITION",
      "condition": {
        "condition_type": "OR",
        "statements": [
          {
              "left": "${L:A:IPv}",
              "op": "==",
              "right": "6"
          }
        ],
        "next": "Get IPv6Fixed _ref"
```

```
        }
    },
    {
        "name": "Get IPv4Fixed _ref",
        "operation": "GET",
        "transport": {
            "path": "fixedaddress?ipv4addr=${L:U:GetIP}&network_view=
${L:U:network_view}&_return_fields=extattrs"
        },
        "wapi": "v2.7"
    },
    {
        "operation": "CONDITION",
        "name": "wapi_response_getIPv4Fix_ref",
        "condition": {
            "condition_type": "AND",
            "statements": [
                {
                    "left": "${P:A:PARSE[0]{_ref}}",
                    "op": "!=",
                    "right": ""
                }
            ],
            "next": "Get_Objref"
        }
    },
    {
        "name": "Get HostIPv4 _ref",
        "operation": "GET",
        "transport": {
            "path": "record:host?ipv4addr=${L:U:GetIP}&network_view=
${L:U:network_view}&_return_fields=extattrs"
        },
        "wapi": "v2.7"
    },
    {
        "operation": "CONDITION",
        "name": "wapi_response_getIPv4Host_ref",
        "condition": {
            "condition_type": "AND",
            "statements": [
                {
```

```
                "left": "${P:A:PARSE[0]{_ref}}",
                "op": "!=",
                "right": ""
            }
        ],
        "next": "Get_Objref"
    }
},
{
    "name": "IPv4 object was not found",
    "operation": "CONDITION",
    "condition": {
        "condition_type": "OR",
        "statements": [
            {
                "left": "1",
                "op": "==",
                "right": "1"
            }
        ],
        "next": "Check if host ipv6"
    }
},
{
    "name": "Get IPv6Fixed _ref",
    "operation": "GET",
    "transport": {
        "path": "ipv6fixedaddress?ipv6addr=${L:U:GetIP}
&network_view=${L:U:network_view}&_return_fields=extattrs"
    },
    "wapi": "v2.7"
},
{
    "operation": "CONDITION",
    "name": "wapi_response_getIPv6Fix_ref",
    "condition": {
        "condition_type": "AND",
        "statements": [
            {
                "left": "${P:A:PARSE[0]{_ref}}",
                "op": "!=",
                "right": ""
```

```
            }
          ],
          "next": "Get_Objref"
        }
      },
      {
        "name": "Get HostIPv6 _ref",
        "operation": "GET",
        "transport": {
          "path": "record:host?ipv6addr=${L:U:GetIP}&network_view=
${L:U:network_view}&_return_fields=extattrs"
        },
        "wapi": "v2.7"
      },
      {
        "operation": "CONDITION",
        "name": "wapi_response_getIPv6Host_ref",
        "condition": {
          "condition_type": "AND",
          "statements": [
            {
              "left": "${P:A:PARSE[0]{_ref}}",
              "op": "!=",
              "right": ""
            }
          ],
          "next": "Get_Objref"
        }
      },
      {
        "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}}}"
```

```json
        }
    },
    {
        "name": "jump if no Obj_ref",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "OR",
            "statements": [
                {
                    "left": "${L:A:Obj_ref}",
                    "op": "==",
                    "right": ""
                }
            ],
            "next": "Check if host ipv6"
        }
    },
    {
        "name": "Update GUID",
        "operation": "PUT",
        "transport": {
            "path": "${L:A:Obj_ref}"
        },
        "wapi": "v2.7",
        "wapi_quoting": "JSON",
        "body_list": [
            "{\"extattrs+\":{\"ePO_GUID\": { \"value\": \"${L:A:GUID}\"},
\"DXL_LastEventSentAt\": { \"value\": \"${E:A:timestamp}\"}}}"
        ]
    },
    {
        "name": "Check if host ipv6",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "AND",
            "statements": [
                {
                    "left": "${E::event_type}",
                    "op": "==",
                    "right": "HOST_ADDRESS_IPV6"
                }
            ],
```

```json
            "next": "GET HOST IPV6 data"
        }
    },
    {
        "name": "GET HOST IPV4 data",
        "operation": "GET",
        "transport": {
            "path": "${E::values{_ref}}?_return_fields=configure_for_dhcp,ho
st,ipv4addr,is_invalid_mac,mac,network,network_view,reserved_interface"¢
            "wapi": "v2.7"
        },
    },
    {
        "name": "Debug#28",
        "operation": "NOP",
        "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
    },
    {
        "name": "set HOST_IPV4 vars",
        "operation": "NOP",
        "body_list": [
            "${XC:COPY:{L:configure_for_dhcp}:{P:configure_for_dhcp}}",
            "${XC:COPY:{L:host}:{P:host}}",
            "${XC:COPY:{L:ipv4addr}:{P:ipv4addr}}",
            "${XC:COPY:{L:is_invalid_mac}:{P:is_invalid_mac}}",
            "${XC:COPY:{L:network_view}:{P:network_view}}",
            "${XC:COPY:{L:reserved_interface}:{P:reserved_interface}}",
            "${XC:ASSIGN:{L:ipv6addr}:{S:}}",
            "${XC:ASSIGN:{L:ipv6prefix}:{S:}}",
            "${XC:ASSIGN:{L:discover_now_status}:{S:}}",
            "${XC:ASSIGN:{L:domain_name}:{S:}}",
            "${XC:ASSIGN:{L:duid}:{S:}}",
            "${XC:ASSIGN:{L:ipv6prefix_bits}:{S:}}"
        ]
    },
    {
        "name": "Check mac",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "AND",
            "statements": [
```

```
                {
                    "left": "${P::PARSE[0]{mac}}",
                    "op": "!=",
                    "right": ""
                }
            ],
            "eval": "${XC:COPY:{L:mac}:{P:PARSE[0]{mac}}}",
            "else_eval": "${XC:ASSIGN:{L:mac}:{S:}}"
        }
    },
    {
        "name": "Check network ",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "AND",
            "statements": [
                {
                    "left": "${P::PARSE[0]{network}}",
                    "op": "!=",
                    "right": ""
                }
            ],
            "eval": "${XC:COPY:{L:network }:{P:PARSE[0]{network}}}",
            "else_eval": "${XC:ASSIGN:{L:network}:{S:}}"
        }
    },
    {
        "name": "Debug#29",
        "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 rest of ipv4 data",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "AND",
            "statements": [
                {
                    "left": "1",
                    "op": "==",
```

```
                    "right": "1"
                }
            ],
            "next": "GET General HOST data IPv4"
        }
    },
    {
        "name": "Debug#30",
        "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 HOST IPV6 data",
        "operation": "GET",
        "transport": {
            "path": "${E::values{_ref}}?_return_fields=configure_for_dhcp,di
scover_now_status,duid,host,ipv6prefix_bits,domain_name,ipv6addr,ipv6p
refix,network_view,reserved_interface"¢         ÒÊ
            "wapi": "v2.7"
    },
    {
        "name": "Debug#31",
        "operation": "NOP",
        "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
    },
    {
        "name": "set HOST_IPV6 vars",
        "operation": "NOP",
        "body_list": [
            "${XC:COPY:{L:configure_for_dhcp}:{P:configure_for_dhcp}}",
            "${XC:COPY:{L:discover_now_status}:
{P:discover_now_status}}",
            "${XC:COPY:{L:host}:{P:host}}",
            "${XC:COPY:{L:ipv6addr}:{P:ipv6addr}}",
            "${XC:COPY:{L:ipv6prefix}:{P:ipv6prefix}}",
            "${XC:COPY:{L:network_view}:{P:network_view}}",
            "${XC:COPY:{L:reserved_interface}:{P:reserved_interface}}",
            "${XC:ASSIGN:{L:ipv4addr}:{S:}}",
```

```
      "${XC:ASSIGN:{L:is_invalid_mac}:{S:}}",
      "${XC:ASSIGN:{L:mac}:{S:}}",
      "${XC:ASSIGN:{L:network}:{S:}}"
    ]
  },
  {
    "name": "Check domain_name",
    "operation": "CONDITION",
    "condition": {
      "condition_type": "AND",
      "statements": [
        {
          "left": "${P::PARSE[0]{domain_name}}",
          "op": "!=",
          "right": ""
        }
      ],
      "eval": "${XC:COPY:{L:domain_name}:{P:PARSE[0]
{domain_name}}}",
      "else_eval": "${XC:ASSIGN:{L:domain_name}:{S:}}"
    }
  },
  {
    "name": "Check duid",
    "operation": "CONDITION",
    "condition": {
      "condition_type": "AND",
      "statements": [
        {
          "left": "${P::PARSE[0]{duid}}",
          "op": "!=",
          "right": ""
        }
      ],
      "eval": "${XC:COPY:{L:duid}:{P:PARSE[0]{duid}}}",
      "else_eval": "${XC:ASSIGN:{L:duid}:{S:}}"
    }
  },
  {
    "name": "Check ipv6prefix_bits",
    "operation": "CONDITION",
    "condition": {
```

```
          "condition_type": "AND",
          "statements": [
            {
              "left": "${P::PARSE[0]{ipv6prefix_bits}}",
              "op": "!=",
              "right": ""
            }
          ],
          "eval": "${XC:COPY:{L:ipv6prefix_bits}:{P:PARSE[0]
{ipv6prefix_bits}}}",
          "else_eval": "${XC:ASSIGN:{L:ipv6prefix_bits}:{S:}}"
        }
      },
      {
        "name": "Debug#32",
        "operation": "NOP",
        "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
      },
      {
        "name": "Get rest of ipv6 data",
        "operation": "CONDITION",
        "condition": {
          "condition_type": "AND",
          "statements": [
            {
              "left": "1",
              "op": "==",
              "right": "1"
            }
          ],
          "next": "GET General HOST data IPv6"
        }
      },
      {
        "name": "GET General HOST data IPv4",
        "operation": "GET",
        "transport": {
          "path": "record:host?name=${E::values{host}}&ipv4addr=
${E::values{ipv4addr}}&_return_fields=allow_telnet,comment,configure_for
_dns,ddns_protected,disable,disable_discovery,dns_name,extattrs,name,
```

network_view,rrset_order,view,zone"¢   ÒÊ
        "wapi": "v2.7"
    },
    {
        "name": "Debug#33",
        "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 General HOST data vars IPv4",
        "operation": "NOP",
        "body_list": [
            "${XC:COPY:{L:allow_telnet}:{P:PARSE[0]{allow_telnet}}}",
            "${XC:COPY:{L:configure_for_dns}:{P:PARSE[0]
{configure_for_dns}}}",
            "${XC:COPY:{L:ddns_protected}:{P:PARSE[0]
{ddns_protected}}}",
            "${XC:COPY:{L:disable}:{P:PARSE[0]{disable}}}",
            "${XC:COPY:{L:disable_discovery}:{P:PARSE[0]
{disable_discovery}}}",
            "${XC:COPY:{L:dns_name}:{P:PARSE[0]{dns_name}}}",
            "${XC:COPY:{L:extattrs}:{P:PARSE[0]{extattrs}{ePO_GUID}
{value}}}",
            "${XC:COPY:{L:name}:{P:PARSE[0]{name}}}",
            "${XC:COPY:{L:network_view}:{P:PARSE[0]{network_view}}}",
            "${XC:COPY:{L:rrset_order}:{P:PARSE[0]{rrset_order}}}",
            "${XC:COPY:{L:view}:{P:PARSE[0]{view}}}"
        ]
    },
    {
        "name": "Debug#34",
        "operation": "NOP",
        "body": "${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": "Check comment ipv4",
        "operation": "CONDITION",
        "condition": {

```
          "condition_type": "AND",
          "statements": [
            {
              "left": "${P::PARSE[0]{comment}}",
              "op": "!=",
              "right": ""
            }
          ],
          "eval": "${XC:COPY:{L:comment}:{P:PARSE[0]{comment}}}",
          "else_eval": "${XC:ASSIGN:{L:comment}:{S:}}"
        }
      },
      {
        "name": "Skip to send Data",
        "operation": "CONDITION",
        "condition": {
          "condition_type": "AND",
          "statements": [
            {
              "left": "1",
              "op": "==",
              "right": "1"
            }
          ],
          "next": "check DXL_MessageFormat"
        }
      },
      {
        "name": "GET General HOST data IPv6",
        "operation": "GET",
        "transport": {
          "path": "record:host?name=${E::values{host}}&ipv6addr=
${E::values{ipv6addr}}&_return_fields=allow_telnet,comment,configure_for
_dns,ddns_protected,disable,disable_discovery,dns_name,extattrs,name,
network_view,rrset_order,view,zone"¢          ÒÊ
          "wapi": "v2.7"
        },
      {
        "name": "Debug#36",
        "operation": "NOP",
        "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
```

```
{UT:}}${XC:DEBUG:{R:}}"
    },
    {
        "name": "set General HOST data vars IPv6",
        "operation": "NOP",
        "body_list": [
            "${XC:COPY:{L:allow_telnet}:{P:PARSE[0]{allow_telnet}}}",
            "${XC:COPY:{L:configure_for_dns}:{P:PARSE[0]
{configure_for_dns}}}",
            "${XC:COPY:{L:ddns_protected}:{P:PARSE[0]
{ddns_protected}}}",
            "${XC:COPY:{L:disable}:{P:PARSE[0]{disable}}}",
            "${XC:COPY:{L:disable_discovery}:{P:PARSE[0]
{disable_discovery}}}",
            "${XC:COPY:{L:dns_name}:{P:PARSE[0]{dns_name}}}",
            "${XC:COPY:{L:extattrs}:{P:PARSE[0]{extattrs}{ePO_GUID}
{value}}}",
            "${XC:COPY:{L:name}:{P:PARSE[0]{name}}}",
            "${XC:COPY:{L:network_view}:{P:PARSE[0]{network_view}}}",
            "${XC:COPY:{L:rrset_order}:{P:PARSE[0]{rrset_order}}}",
            "${XC:COPY:{L:view}:{P:PARSE[0]{view}}}"
        ]
    },
    {
        "name": "Check comment ipv6",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "AND",
            "statements": [
                {
                    "left": "${P::PARSE[0]{comment}}",
                    "op": "!=",
                    "right": ""
                }
            ],
            "eval": "${XC:COPY:{L:comment}:{P:PARSE[0]{comment}}}",
            "else_eval": "${XC:ASSIGN:{L:comment}:{S:}}"
        }
    },
    {
        "name": "check DXL_MessageFormat",
        "operation": "CONDITION",
```

```
        "condition": {
          "condition_type": "AND",
          "statements": [
            {
              "left": "${I::DXL_MessageFormat}",
              "op": "==",
              "right": "CEF"
            }
          ],
          "next": "send_CEF"
        }
      },
      {
        "name": "Debug#37",
        "operation": "NOP",
        "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
      },
      {
        "name": "send_OpenDXL",
        "operation": "DXL_SEND_EVENT",
        "body_list": [
          "{",
          "    \"eventMsgType\": \"Infoblox Change Event\",",
          "    \"eventMsgVersion\": \"1.0\",",
          "    \"event\": {",
          "        \"category\": \"${E::event_type}\",",
          "        \"eventDesc\": \"DNS ${E::event_type}
${E::operation_type} event\",",
          "        \"eventType\": \"${E::operation_type}\",",
          "        \"eventId\": \"204160\",",
          "        \"analyzer\": {",
          "            \"id\": \"S_INFBLX0802\",",
          "            \"version\": \"8.2.1\",",
          "            \"name\": \"NIOS\",",
          "            \"detectionMethod\": \"NIOS\",",
          "            \"hostName\": \"${E::member_name}\",",
          "            \"detectedUTC\": \"${L::DetectedUTC}\",",
          "            \"ipv4\": \"${L::internal{analyzer_ipv4}}\",",
          "            \"ipv6\": \"${L::internal{analyzer_ipv6}}\"",
          "        },",
```

"   \"entity\": {",
"   \"groupName\": \"\",",
"   \"osPlatform\": \"\",",
"   \"osType\": \"\",",
"   \"type\": \"\",",
"   \"sessionID\": \"\",",
"   \"allow_telnet\": \"${L::allow_telnet}\",",
"   \"configure_for_dns\": \"${L::configure_for_dns}\",",
"   \"ddns_protected\": \"${L::ddns_protected}\",",
"   \"disable\": \"${L::disable}\",",
"   \"disable_discovery\": \"${L::disable_discovery}\",",
"   \"dns_name\": \"${L::dns_name}\",",
"   \"name\": \"${L::name}\",",
"   \"network_view\": \"${L::network_view}\",",
"   \"rrset_order\": \"${L::rrset_order}\",",
"   \"domain_name\": \"${L::domain_name}\",",
"   \"duid\": \"${L::duid}\",",
"   \"mac\": \"${L::mac}\",",
"   \"network \": \"${L::network}\",",
"   \"view\": \"${L::view}\",",
"   \"username\": \"${L::username}\",",
"   \"domainname\": \"${L::domainname}\",",
"   \"comment\": \"${L::comment}\",",
"   \"extattr\":{",
"   \"ePO_GUID\": \"${L::extattrs}\"",
"   },",
"   \"ipv4addr\":{",
"   \"configure_for_dhcp\": \"${L::configure_for_dhcp}\",",
"   \"host\": \"${L::host}\",",
"   \"ipv4addr\": \"${L::ipv4addr}\",",
"   \"is_invalid_mac\": \"${L::is_invalid_mac}\",",
"   \"reserved_interface\": \"${L::reserved_interface}\"",
"   },",
"   \"ipv6addr\":{",
"   \"configure_for_dhcp\": \"${L::configure_for_dhcp}\",",
"   \"discover_now_status\":
\"${L::discover_now_status}\",",
"   \"host\": \"${L::host}\",",
"   \"ipv6prefix_bits \": \"${L::ipv6prefix_bits}\",",
"   \"ipv6addr\": \"${L::ipv6addr}\",",
"   \"ipv6prefix\": \"${L::ipv6prefix}\",",
"   \"reserved_interface\": \"${L::reserved_interface}\"",

```
"               ",
"           }",
"       },",
"         \"source\": {",
"             \"ipv4\": \"${L::internal{source_ipv4}}\",",
"             \"ipv6\": \"${L::internal{source_ipv6}}\",",
"             \"port\": 00000",
"         }",
"     }",
"}"
],
"dxl_topic": "/open/DDI/v1/${E::event_type}/infoblox"
},
{
    "name": "Debug#50",
    "operation": "NOP",
    "body": "${XC:DEBUG:{H:}}${XC:DEBUG:{E:}}${XC:DEBUG:
{I:}}${XC:DEBUG:{L:}}${XC:DEBUG:{S:}}${XC:DEBUG:{P:}}${XC:DEBUG:
{UT:}}${XC:DEBUG:{R:}}"
},
{
    "name": "goFin",
    "operation": "CONDITION",
    "condition": {
        "condition_type": "OR",
        "statements": [
            {
                "left": "1",
                "op": "==",
                "right": "1"
            }
        ],
        "next": "Fin"
    }
},
{
    "name": "send_CEF",
    "operation": "DXL_SEND_EVENT",
    "body_list": [
        "{\"DXLCommonEvent\":{",
        "\"category\": \"${E::event_type}\",",
        "\"eventDesc\": \"DNS ${E::event_type} ${E::operation_type}
```

event\",",
   \"eventType\": \"${E::operation_type}\",",
   \"eventId\": \"204160\",",
   \"AgentGUID\": \"${L::GUID}\",",
   \"Analyzer\":
\"${L::internal{analyzer_ipv4}}${L::internal{analyzer_ipv6}}\",",
   \"AnalyzerDATVersion\": \"\",",
   \"AnalyzerDetectionMethod\": \"${E::object_type}\",",
   \"AnalyzerHostName\": \"${E::member_name}\",",
   \"AnalyzerIPV4\": \"${L::internal{analyzer_ipv4}}\",",
   \"AnalyzerIPV6\": \"${L::internal{analyzer_ipv6}}\",",
   \"AnalyzerMAC\": \"\",",
   \"AnalyzerName\": \"NIOS\",",
   \"AnalyzerVersion\": \"8.2.1\",",
   \"DetectedUTC\": \"${L::DetectedUTC}\",",
   \"ServerID\":
\"${L::internal{analyzer_ipv4}}${L::internal{analyzer_ipv6}}\",",
   \"SourceIPV4\": \"${L::internal{source_ipv4}}\",",
   \"SourceIPV6\": \"${L::internal{source_ipv6}}\",",
   \"SourcePort\": \"00000\",",
   \"TargetHostName\": \"${E::member_name}\",",
   \"TargetIPV4\": \"${L::internal{analyzer_ipv4}}\",",
   \"TargetIPV6\": \"${L::internal{analyzer_ipv6}}\",",
   \"TargetPort\": \"53\",",
   \"TargetProtocol\": \"dns\",",
   \"allow_telnet\": \"${L::allow_telnet}\",",
   \"configure_for_dns\": \"${L::configure_for_dns}\",",
   \"ddns_protected\": \"${L::ddns_protected}\",",
   \"disable\": \"${L::disable}\",",
   \"disable_discovery\": \"${L::disable_discovery}\",",
   \"dns_name\": \"${L::dns_name}\",",
   \"name\": \"${L::name}\",",
   \"network_view\": \"${L::network_view}\",",
   \"rrset_order\": \"${L::rrset_order}\",",
   \"view\": \"${L::view}\",",
   \"ePO_GUID\": \"${L::extattrs}\"",
   \"username\": \"${L::username}\",",
   \"domainname\": \"${L::domainname}\",",
   \"configure_for_dhcp\": \"${L::configure_for_dhcp}\",",
   \"host\": \"${L::host}\",",
   \"ipv4addr\": \"${L::ipv4addr}\",",
   \"is_invalid_mac\": \"${L::is_invalid_mac}\",",

```
            \"reserved_interface\": \"${L::reserved_interface}\"",
            \"discover_now_status\": \"${L::discover_now_status}\",",
            \"ipv6addr\": \"${L::ipv6addr}\",",
            \"ipv6prefix\": \"${L::ipv6prefix}\",",
            \"domain_name\": \"${L::domain_name}\",",
            \"duid\": \"${L::duid}\",",
            \"ipv6prefix_bits\": \"${L::ipv6prefix_bits}\",",
            \"mac\": \"${L::mac}\",",
            \"comment\": \"${L::comment}\",",
            \"network \": \"${L::network}\"",
            "}}"
         ],
         "dxl_topic": "/infoblox/outbound/${E::event_type}"
    },
    {
        "name": "goFin#2",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "OR",
            "statements": [
                {
                    "left": "1",
                    "op": "==",
                    "right": "1"
                }
            ],
            "next": "Fin"
        }
    },
    {
        "name": "check DXL_MessageFormat_Delete",
        "operation": "CONDITION",
        "condition": {
            "condition_type": "AND",
            "statements": [
                {
                    "left": "${I::DXL_MessageFormat}",
                    "op": "==",
                    "right": "CEF"
                }
            ],
            "next": "send_CEF_Delete"
```

```
            }
        },
        {
            "name": "send_OpenDXL_Delete",
            "operation": "DXL_SEND_EVENT",
            "body_list": [
                "{",
                "    \"eventMsgType\": \"Infoblox Change Event\",",
                "    \"eventMsgVersion\": \"1.0\",",
                "    \"event\": {",
                "        \"category\": \"${E::event_type}\",",
                "        \"eventDesc\": \"DNS ${E::event_type}
${E::operation_type} event\",",
                "        \"eventType\": \"${E::operation_type}\",",
                "        \"eventId\": \"204160\",",
                "        \"analyzer\": {",
                "            \"id\": \"S_INFBLX0802\",",
                "            \"version\": \"8.2.1\",",
                "            \"name\": \"NIOS\",",
                "            \"detectionMethod\": \"NIOS\",",
                "            \"hostName\": \"${E::member_name}\",",
                "            \"detectedUTC\": \"${L::DetectedUTC}\",",
                "            \"ipv4\": \"${L::internal{analyzer_ipv4}}\",",
                "            \"ipv6\": \"${L::internal{analyzer_ipv6}}\"",
                "        },",
                "        \"source\": {",
                "            \"ipv4\": \"${L::internal{source_ipv4}}\",",
                "            \"ipv6\": \"${L::internal{source_ipv6}}\",",
                "            \"port\": 00000",
                "        }",
                "    }",
                "}"
            ],
            "dxl_topic": "/open/DDI/v1/${E::event_type}/infoblox"
        },
        {
            "name": "goFin#3",
            "operation": "CONDITION",
            "condition": {
                "condition_type": "OR",
                "statements": [
                    {
```

```json
              "left": "1",
              "op": "==",
              "right": "1"
            }
          ],
          "next": "Fin"
        }
    },
    {
        "name": "send_CEF_Delete",
        "operation": "DXL_SEND_EVENT",
        "body_list": [
          "{\"DXLCommonEvent\":{",
          "\"category\": \"${E::event_type}\",",
          "\"eventDesc\": \"DNS ${E::event_type} ${E::operation_type} event\",",
          "\"eventType\": \"${E::operation_type}\",",
          "\"eventId\": \"204160\",",
          "\"AgentGUID\": \"${L::GUID}\",",
          "\"Analyzer\": \"${L::internal{analyzer_ipv4}}${L::internal{analyzer_ipv6}}\",",
          "\"AnalyzerDATVersion\": \"\",",
          "\"AnalyzerDetectionMethod\": \"${E::object_type}\",",
          "\"AnalyzerHostName\": \"${E::member_name}\",",
          "\"AnalyzerIPV4\": \"${L::internal{analyzer_ipv4}}\",",
          "\"AnalyzerIPV6\": \"${L::internal{analyzer_ipv6}}\",",
          "\"AnalyzerMAC\": \"\",",
          "\"AnalyzerName\": \"NIOS\",",
          "\"AnalyzerVersion\": \"8.2.1\",",
          "\"DetectedUTC\": \"${L::DetectedUTC}\",",
          "\"ServerID\": \"${L::internal{analyzer_ipv4}}${L::internal{analyzer_ipv6}}\",",
          "\"SourceIPV4\": \"${L::internal{source_ipv4}}\",",
          "\"SourceIPV6\": \"${L::internal{source_ipv6}}\",",
          "\"SourcePort\": \"00000\",",
          "\"TargetHostName\": \"${E::member_name}\",",
          "\"TargetIPV4\": \"${L::internal{analyzer_ipv4}}\",",
          "\"TargetIPV6\": \"${L::internal{analyzer_ipv6}}\",",
          "\"TargetPort\": \"53\",",
          "\"TargetProtocol\": \"dns\"",
          "}}"
        ],
```

```
      "dxl_topic": "/infoblox/outbound/${E::event_type}"
    },
    {
      "name": "Fin",
      "operation": "NOP",
      "body": ""
    }
  ]
}
```