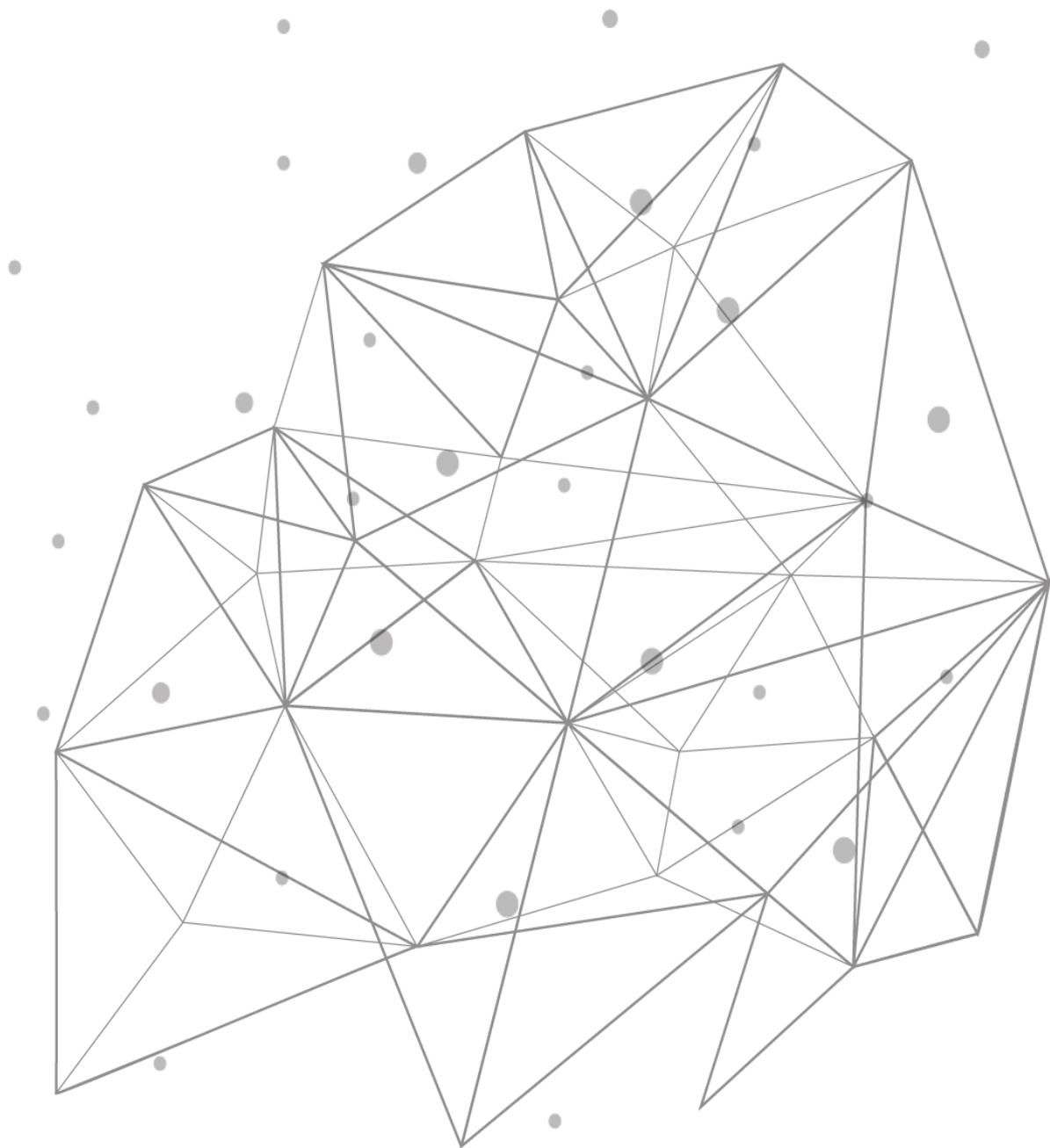

ServiceNow Integration



TCPWave® Inc

600 Alexander Road

Princeton, NJ 08540

USA

Phone: 888-831-8276

Email: support@tcpwave.com

Website: www.tcpwave.com

This document is the proprietary and confidential property of TCPWave® Inc. All resulting rights and translation and duplication rights are reserved and subject to a separate agreement. Do not share without prior approval.

TCPWave® Inc. reserves the right to modify the described product in compliance with technical progress at any time and without prior notice unless otherwise provided in the agreement.

Table of Contents

Introduction	4
TCPWave Integration with ServiceNow	4
Architecture Overview	4
ServiceNow Versions Supported by TCPWave	5
Configuration of TCPWave and ServiceNow	5
Configuring Authentication	5
Configuring Script Include Functions in ServiceNow	7
Configuring Workflows in ServiceNow	8
Defining Change Requests in ServiceNow	10
Resulting Workflow Output.....	10
Workflows Supported by TCPWave.....	11
Advantages of ServiceNow Integration	12
Appendix: Example Values in Change Requests for DDI Operations.....	13
Network Add	13
Network Delete	14
Zone Resource Record Add for “A” Record	14
Zone Resource Record Delete for “A” Record.....	14
Zone Resource Record Add for CNAME Record	15
Zone Resource Record Delete for CNAME Record	15
DHCP Scope Add.....	16
DHCP Scope Delete	16
DHCP Manual Object Add	17
DHCP Manual Object Delete	17
IPAM Static Object Add	18
IPAM Static Object Delete	18

Introduction

Enterprises that use TCPWave IPAM can leverage seamless integration with ServiceNow workflows. Activities, such as Change Request Approval, Reporting an Incident, and Planning a Corrective Plan of Action, that are performed in a workflow in ServiceNow can invoke Script Include functions. These functions can securely make API calls to TCPWave IPAM to invoke DDI functionality in it.

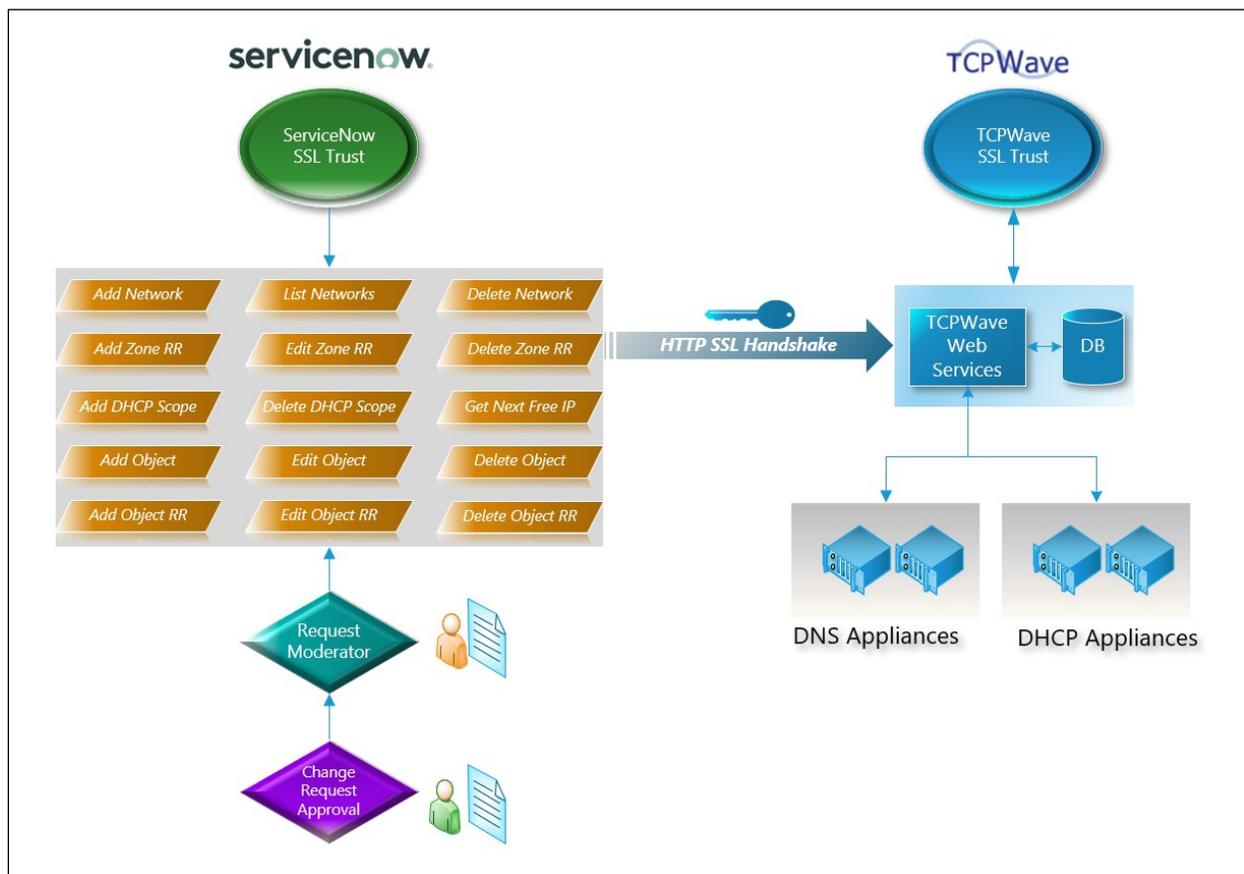
To enable users to understand and take advantage of TCPWave’s integration with ServiceNow, information on the following topics is presented below:

- Architecture and overview of TCPWave and ServiceNow integration
- Configuration of TCPWave and ServiceNow needed for integration
- Workflows in ServiceNow that TCPWave supports
- Example values in Change Requests in ServiceNow

TCPWave Integration with ServiceNow

Architecture Overview

The architecture diagram below provides an overview of how user actions in a workflow in ServiceNow can invoke REST API calls to TCPWave IPAM. ServiceNow and TCPWave IPAM communicate securely over the API using HTTPS (HTTP with SSL).



As an example, a summary of the main steps in a workflow for a Change Request to add a DNS domain in TCPWave IPAM is provided below. The Description field in the Change Request contains a JSON payload

that contains the attributes and associated values that are needed to make the change in TCPWave IPAM.

1. A user approves a Change Request in the Change Request Approval activity.
2. In the Request Moderator activity, a Script Include function is invoked for an Add DNS Domain action.
3. The Request Moderator activity initiates an API REST call to TCPWave IPAM to add the domain.
4. ServiceNow and TCPWave IPAM perform authentication by doing an SSL handshake and exchanging certificates in their key stores.
5. TCPWave IPAM accepts the request to add the DNS domain and adds it.

ServiceNow Versions Supported by TCPWave

The following versions of ServiceNow are supported by TCPWave:

- Jakarta
- Kingston
- London
- Madrid
- New York
- Orlando

Configuration of TCPWave and ServiceNow

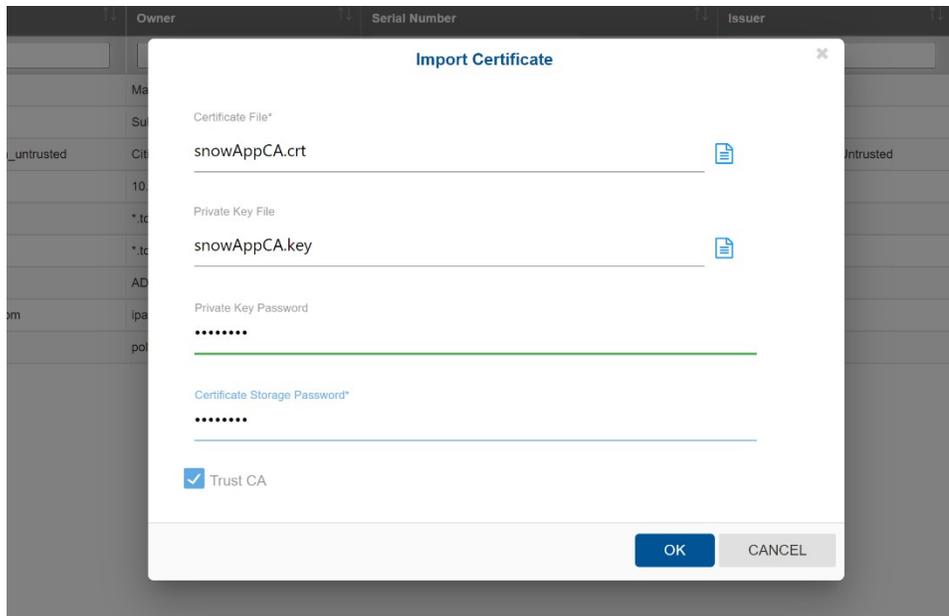
To have ServiceNow invoke functionality in TCPWave, first configure authentication in TCPWave and ServiceNow. Then configure Script Include functions, workflows, and Change Requests in ServiceNow.

Configuring Authentication

An example of configuring authentication in TCPWave and ServiceNow using self-signed SSL certificates is presented in the steps below. However, for improved security, TCPWave highly recommends using valid certificates signed by trusted authorities.

1. Create a root certificate by using the following commands:

```
openssl genrsa -des3 -out snowAppCA.key 4096
openssl req -x509 -new -nodes -key snowAppCA.key -sha256 -
days 1024 -out snowAppCA.crt
```
2. In TCPWave IPAM, import the files snowAppCA.crt and snowAppCA.key created in the previous step as a certificate for an appliance. When doing the import, ensure that **Trust CA** is selected as shown in the screenshot below.



3. Create a user certificate by using the following commands:

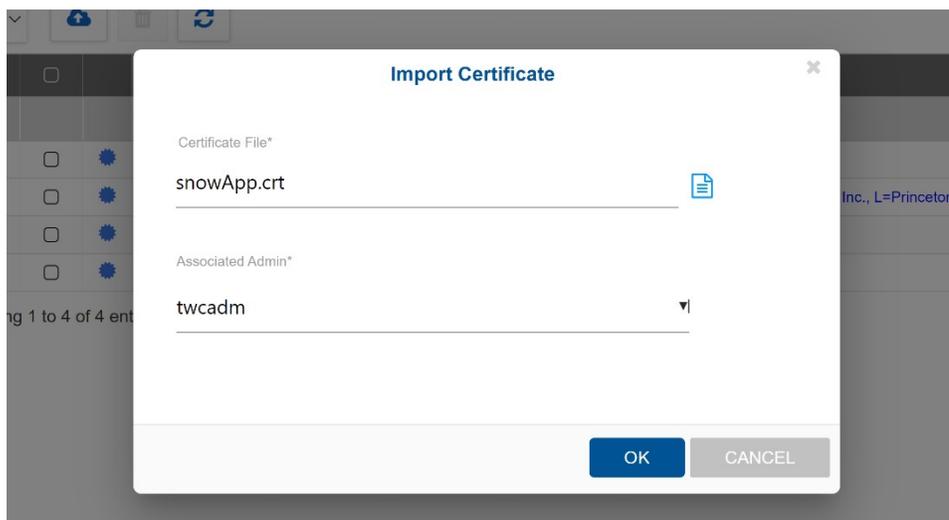
```
openssl genrsa -out snowApp.key 2048
```

```
openssl req -new -key snowApp.key -out snowApp.csr
```

4. Sign the user certificate by root CA by using the following command:

```
openssl x509 -req -in snowApp.csr -CA snowAppCA.crt -CAkey snowAppCA.key -CAcreateserial -out snowApp.crt -days 500 -sha256
```

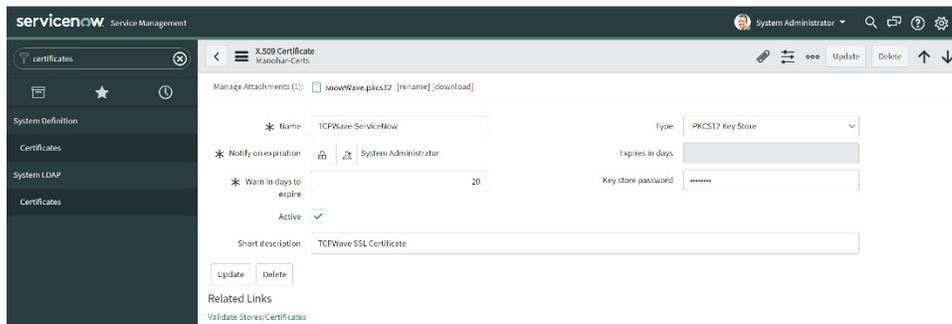
5. Import the user certificate in the file snowApp.crt created in the previous step into TCPWave IPAM, as shown in the screenshot below. This step is required to associate all incoming HTTP requests that use this certificate with an IPAM user.



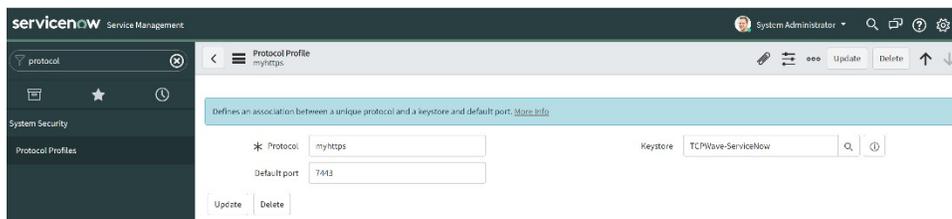
6. Generate a certificate in PKCS12 format by using the following command:

```
openssl pkcs12 -export -in snowApp.crt -inkey snowApp.key -
name snowWave -out snowWave.pkcs12
```

- In ServiceNow, import the TCPWave SSL Certificate in the file snowWave.pkcs12 created in the previous step, as shown in the screenshot below.



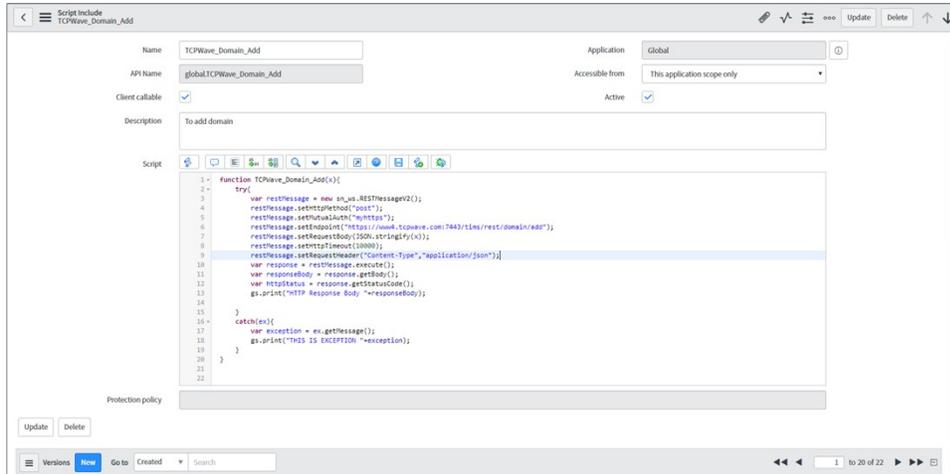
- In ServiceNow, define an association between a unique protocol and a key store and default port, as shown in the screenshot below.



Configuring Script Include Functions in ServiceNow

In ServiceNow, Script Include functions can be created to make API calls to TCPWave IPAM to invoke DDI functionality in it. These functions are runnable JavaScript functions that can be created in the ServiceNow web UI. They can use the native ServiceNow JavaScript API to perform executable actions, such as invoking outbound HTTP REST calls. Consequently, these functions can perform various RESTful actions in TCPWave IPAM.

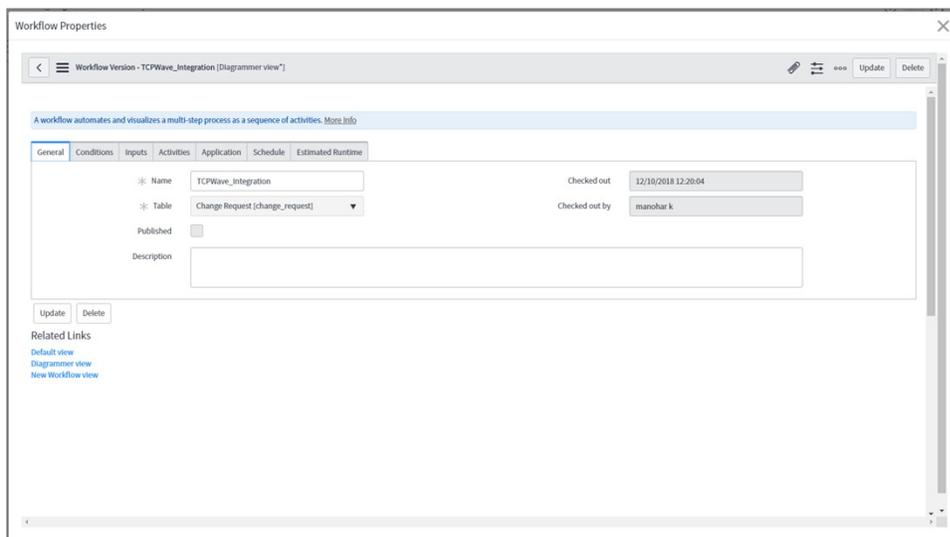
The screenshot below shows an example Script Include function in ServiceNow for adding a domain in TCPWave IPAM. Additional example JavaScript code snippets are provided in the TCPWave Git Repository. You can use these examples as a basis to create the Script Include functions that you need.



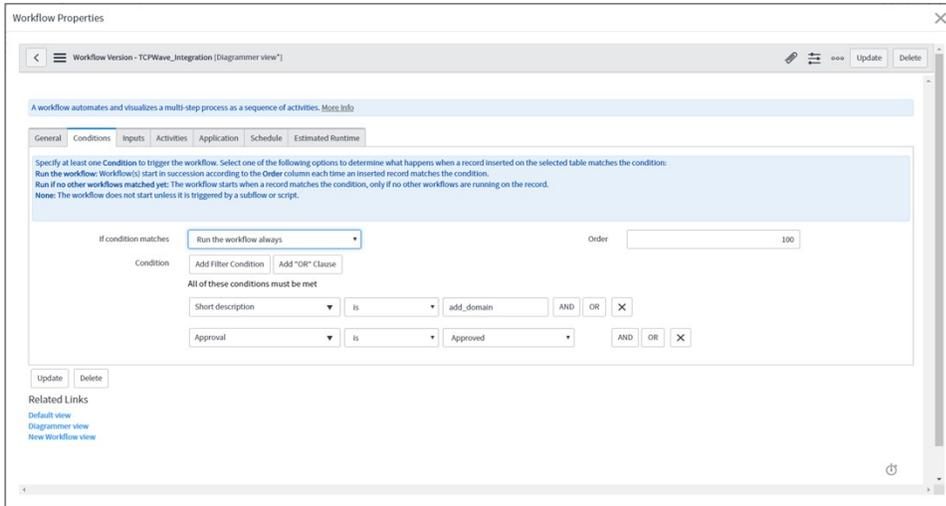
Configuring Workflows in ServiceNow

Workflows in ServiceNow can be configured to be invoked on demand based on various ongoing activities in ServiceNow, such as the approval or implementation of a Change Request. TCPWave IPAM integration with change events in ServiceNow is done by using workflow transitions that use JavaScript directives to retrieve Change Request information and invoke Script Include functions.

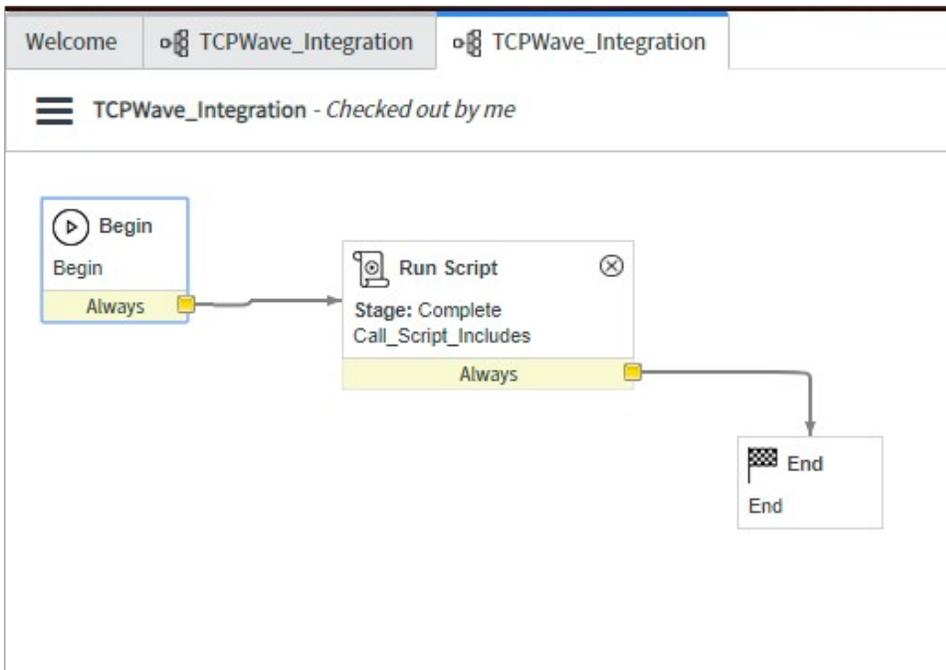
As an example of the configuration of a workflow in ServiceNow, the configuration of a simple one named TCPWave_Integration is presented in subsequent screenshots. You can use this example as a basis to create the workflows that you need. For TCPWave_Integration, general information on it is shown in the screenshot below, including that it operates on the Change Request Table.



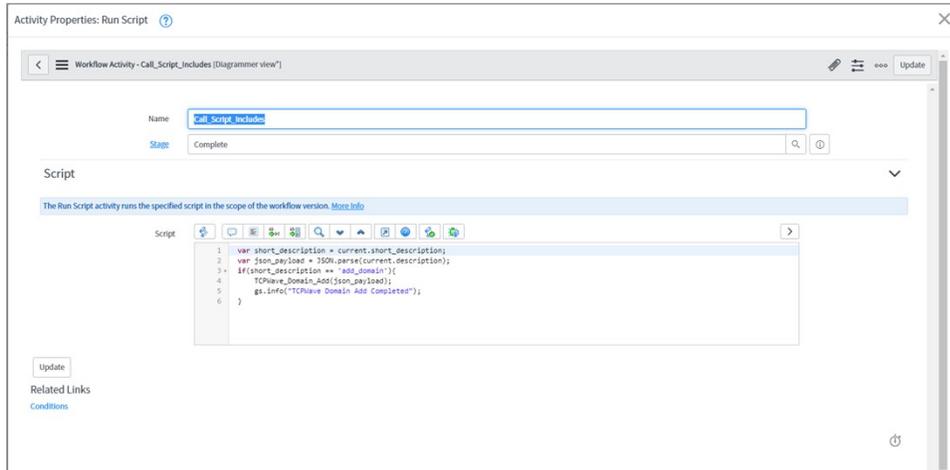
This workflow runs when two conditions are met: the Short Description is add_domain and the Approval status is Approved. These conditions are shown in the following screenshot.



The TCPWave_Integration workflow is shown in the figure below. The activity named Run Script in the workflow invokes a script that retrieves information from a Change Request. This information includes the type of Change Request, which is in the Short Description field, and the JSON payload, which is in the Description field. Example values for Short Description are add_domain and add_static_object.



The script that Run Script runs is shown in the screenshot below. As seen in this script, it checks the Short Description field for the value add_domain in a Change Request and then invokes an associated Script Include function to add the domain.



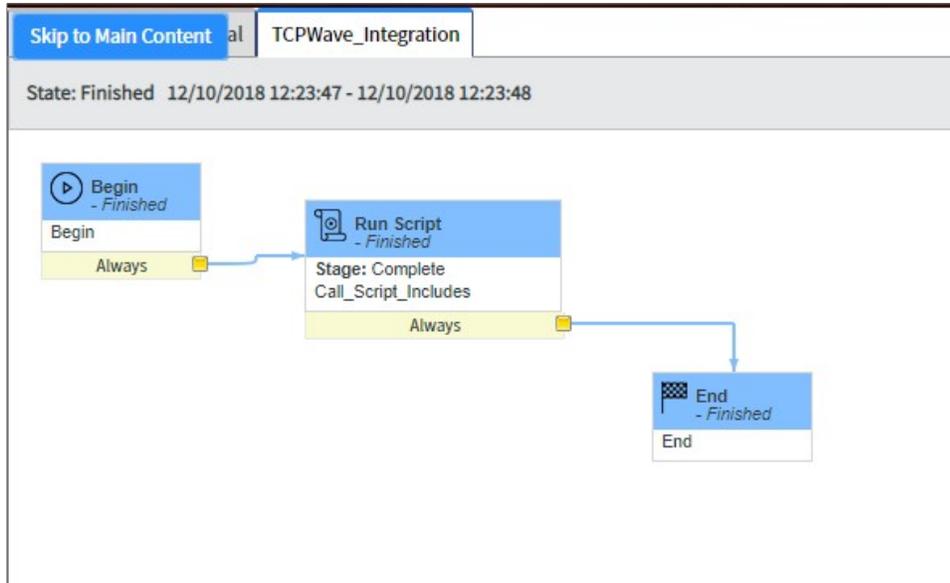
Defining Change Requests in ServiceNow

An example of part of the definition of a Change Request in ServiceNow is shown in the screenshot below. In it, a value that identifies the type of request is specified in the Short Description field. Also, a JSON payload that contains the attributes and associated values that are needed to make the change in TCPWave IPAM is specified in the Description field. Additional example values for these two fields are presented in the appendix, and you can use these examples as a basis to define the Change Requests that you need.



Resulting Workflow Output

After an authorized administrator approves a Change Request in ServiceNow, the configured workflow for it runs. The following figure shows the successful execution of a version of the TCPWave_Integration workflow that was modified to add a static object.



As a result of completing the workflow, a static object named AWS00001Instance.aaa.com with an Object Type of AWS Instance was created in TCPWave IPAM, as shown in the screenshot below..

Address	Name	Domain	Object Type	Created By	Created Time	Updated By	Updated Time	Allocation Type	MAC	Serial Number	Time to Live
99.4.0.1	router-95245	aaa.com	Router	lscadm	09/19/37 10-15-2018			Static			
99.4.0.2								Unallocated			
99.4.0.3								Unallocated			
99.4.0.4								Unallocated			
99.4.0.5								Unallocated			
99.4.0.6	AWS00001Instance	aaa.com	AWS Instance	lscadm	10/12/20 10-15-2018	lscadm	10/22/20 10-15-2018	Static			1200

Workflows Supported by TCPWave

TCPWave supports the workflows below. Example values for the Short Description and Description fields in Change Requests in ServiceNow for some of these workflows are presented in the appendix.

- Network
 - Create Network
 - Delete Network
 - List Networks
- DNS
 - Create “A” Record
 - Delete “A” Record
 - Create CNAME Record
 - Delete CNAME Record
- DHCP
 - Create Scope
 - Delete Scope
 - Create DHCP Manual Object
 - Delete DHCP Manual Object

- IPAM
 - Get Next Free Available IP
 - Create Static Object
 - Delete Static Object
 - Edit Object
 - Delete Object
 - Add Object RR
 - Edit Object RR
 - Delete Object RR
 - Add Zone RR
 - Edit Zone RR
 - Delete Zone RR

Advantages of ServiceNow Integration

The many web services offered by TCPWave DDI can be extensively leveraged from ServiceNow (and similar applications) to quickly, securely, and automatically perform Change Request management and needed DDI activities. Also, the SSL-based authentication and encrypted data exchange used by TCPWave and ServiceNow ensure a trusted connection between them. Enterprises that use TCPWave DDI can seamlessly integrate it with ServiceNow and create custom workflows to meet nearly any need to accomplish safe and secure DDI workflow automation.

Appendix: Example Values in Change Requests for DDI Operations

Example values for the Short Description and Description fields in Change Requests in ServiceNow are presented in the sections below. Each section is for a specific type of DDI operation. The Short Description field contains a value for the type of Change Request, which corresponds to the type of DDI operation. The Description field contains a JSON payload that contains the attributes and associated values needed to make the change in TCPWave IPAM.

These sections are organized by DDI topic in the following sequence:

1. Network
2. Zone Resource Record
3. DHCP Scope
4. DHCP Manual Object
5. IPAM Static Object

Network Add

Short description field: - add_network

Description Field: Copy the JSON payload below

```
{
  "address": "20.0.0.0",
  "mask_length": "24",
  "organization_name": "Internal",
  "name": "ServiceNow Network",
  "description": "SNOW",
  "createRevZone": "yes",
  "dmzVisible": "no",
  "dnssec_enable": "no",
  "nsec_option": "NSEC3",
  "monitoringService": "no",
  "enable_discovery": "no",
  "discovery_template": "",
  "percentageFull": 100,
  "email_check": 1,
  "snmp_check": 0,
  "log_check": 0,
  "zoneTemplateId": "",
  "zoneTemplateName": null,
  "addr1": "20",
  "addr2": "0",
  "addr3": "0",
  "addr4": "0",
  "extensions": []
}
```

Network Delete

Short description field: - del_network

Description Field: Copy the JSON payload below

```
{
  "address": "20.0.0.0",
  "organization_name": "Internal"
}
```

Zone Resource Record Add for "A" Record

Short description field: - add_rr

Description Field: Copy the JSON payload below

```
{
  "zoneName": "snow.com",
  "owner": "ARecord",
  "rrclass": "IN",
  "rrtype": "A",
  "ttl": "1200",
  "data": "10.0.0.2",
  "description": "",
  "is_external_rr": 0,
  "status": 1,
  "organization_name": "Internal"
}
```

Zone Resource Record Delete for "A" Record

Short description field: - del_rr

Description Field: Copy the JSON payload below

```
{
  "zoneName": "snow.com",
  "organization_name": "Internal",
  "rrtype": "A",
  "rrclass": "IN",
  "owner": "ARecord.snow.com.",
  "data": "10.0.0.2"
}
```

Zone Resource Record Add for CNAME Record

Short description field: - add_rr

Description Field: Copy the JSON payload below

```
{
  "zoneName": "snow.com",
  "owner": "CRecord",
  "rrclass": "IN",
  "rrtype": "CNAME",
  "ttl": "1200",
  "data": "ARecord.snow.com.",
  "description": "",
  "is_external_rr": 0,
  "status": 1,
  "organization_name": "Internal"
}
```

Zone Resource Record Delete for CNAME Record

Short description field: - del_rr

Description Field: Copy the JSON payload below

```
{
  "zoneName": "snow.com",
  "organization_name": "Internal",
  "rrtype": "CNAME",
  "rrclass": "IN",
  "owner": "CRecord.snow.com.",
  "data": "ARecord.snow.com."
}
```

DHCP Scope Add

Short description field: - add_scope

Description Field: Copy the JSON payload below

```
{
  "scope": {
    "addressRanges": [{
      "startIP": "1.0.0.3",
      "endIP": "1.0.0.10"
    }],
    "allocation_type": "dynamic",
    "allowClassesArray": [],
    "class_code": "3G Phone",
    "denyClassesArray": [],
    "description": "",
    "organization_name": "Internal",
    "primary_dhcp_server": "DNSAppliance",
    "template_name": "ServiceNow"
  },
  "subnetAddress": "1.0.0.0",
  "update_ns_a": true,
  "update_ns_ptr": true,
  "dyn_update_rrs_a": true,
  "dyn_update_rrs_cname": true,
  "dyn_update_rrs_mx": true,
  "dyn_update_rrs_ptr": true,
  "ttl": 1200
}
```

DHCP Scope Delete

Short description field: - del_scope

Description Field: Copy the JSON payload below

```
{
  "addressRange": "1.0.0.3-1.0.0.10",
  "subnetAddress": "1.0.0.0",
  "organization_name": "Internal"
}
```

DHCP Manual Object Add

Short description field: - add_object

Description Field: Copy the JSON payload below

```
{
  "name": "DHCPManualObject",
  "class_code": "3G Phone",
  "alloc_type": "2",
  "mac": "E1:aa:BB:CC:DD:EE",
  "ttl": "1200",
  "option_template_name": "ServiceNow",
  "dhcp_server": "DHCPAppliance",
  "update_ns_a": true,
  "update_ns_ptr": true,
  "dyn_update_rrs_a": true,
  "dyn_update_rrs_ptr": true,
  "dyn_update_rrs_cname": true,
  "dyn_update_rrs_mx": true,
  "addr1": "1",
  "addr2": "0",
  "addr3": "0",
  "addr4": "11",
  "subnet_address": "1.0.0.0",
  "domain_name": "abc.com",
  "organization_name": "Internal"
}
```

DHCP Manual Object Delete

Short description field: - del_object

Description Field: Copy the JSON payload below

```
{
  "addressArray": ["1.0.0.11"],
  "isDeleterrsChecked": 1,
  "organization_name": "Internal"
}
```

IPAM Static Object Add

Short description field: - add_object

Description Field: Copy the JSON payload below

```
{
  "addr1": 1,
  "addr2": 0,
  "addr3": 0,
  "addr4": 12,
  "name": "CRMServer",
  "alloc_type": 1,
  "class_code": "CRM Server",
  "domain_name": "abc.com",
  "organization_name": "Internal",
  "subnet_address": "1.0.0.0"
}
```

IPAM Static Object Delete

Short description field: - del_object

Description Field: Copy the JSON payload below

```
{
  "addressArray": ["1.0.0.12"],
  "isDeleterrsChecked": 1,
  "organization_name": "Internal"
}
```
