

## Example: Parameter Value Substitution

---

Parameters provide general definitions for configuration properties. You can use parameters in the configuration for policies, services, and subscriptions. Users can define the value for a parameter through an enterprise service portal or a residential portal.



**NOTE:** The SRC sample data includes the configuration used in this example.

---

This example shows how to use parameters and substitutions in the SRC software.

- Requirements on page 1
- Overview on page 1
- Configuration on page 4

### Requirements

This example uses the following hardware and software components:

- SRC software 1.0.0 and greater
- Sample enterprise service portal available with SRC software 1.0.0 and greater
- C-series controller
- JUNOSe router(s)

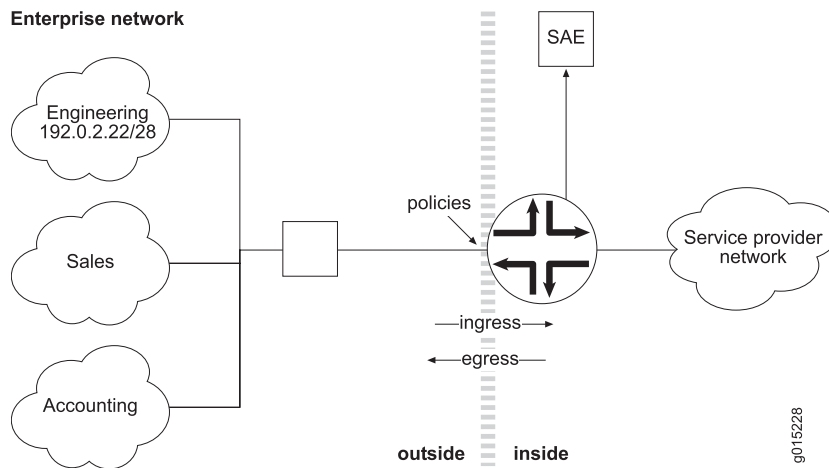
### Overview

This configuration has the following characteristics:

- A service that provides a gold-level quality of service
- A department subnet in an enterprise network subscribes to this service with the ability to track and charge the department for the volume of bandwidth used.

Figure 1 on page 2 shows the network in the example.

**Figure 1: Network Used in Parameter Substitution Example**



From the service provider's perspective, the service provider's network is on the inside, and the enterprise network is on the outside. Ingress traffic flows from the enterprise network to the service provider's network. Egress traffic flows from the service provider's network to the enterprise network. The engineering department subnet in the enterprise network is the subnet that we will subscribe to the gold-level service and track.

## Types of Parameters

The example uses two types of parameters:

- rate—Used to scale the rate limiter
- network—Used to specify IP subnets in classify conditions

## Parameter Configuration

The parameters appear in the configuration for:

- A policy group called `tierpolicy` that classifies packets based on source and destination subnets and applies a rate limit action to those packets. The `tierpolicy` policy group contains three local parameters:
  - `inside`—Parameter of type `network`; used to specify a subnet
  - `outside`—Parameter of type `network`; used to specify a subnet
  - `qos`—Parameter of type `rate`; used to scale the rate limiter
- A service called `GoldMetered`, that has `tierpolicy` as the policy group. The `GoldMetered` service includes the following parameter substitution:
  - `qos`—Fix to 50 % of the `interface_speed` parameter. (`interface_speed` is a global runtime parameter that the SAE fills in with the actual speed of the router interface.)
  - `dept`—Create a parameter called `dept` that is parameter type (role) `network`.

- outside—Set to dept (short for department), which effectively renames the outside parameter to dept.
- inside—Set to any.
- An enterprise subscriber that uses the following parameter substitution:
  - eng—Create a parameter called eng (short for engineering department) that is parameter type (role) network, and set the value to 192.0.2.22/28.
- A subscriber subscription to the GoldMetered service that has the following parameter substitution:
  - dept—Set to eng.

### Parameter Values After Value Acquisition

After the SRC software has gone through the parameter value acquisition process, the three original parameters in the tierpolicy policy group have the following values:

- inside = 0.0.0.0/0

This value was acquired from the global parameter any that was defined in the service definition

- outside = 192.0.2.22/28

This value was acquired as follows:

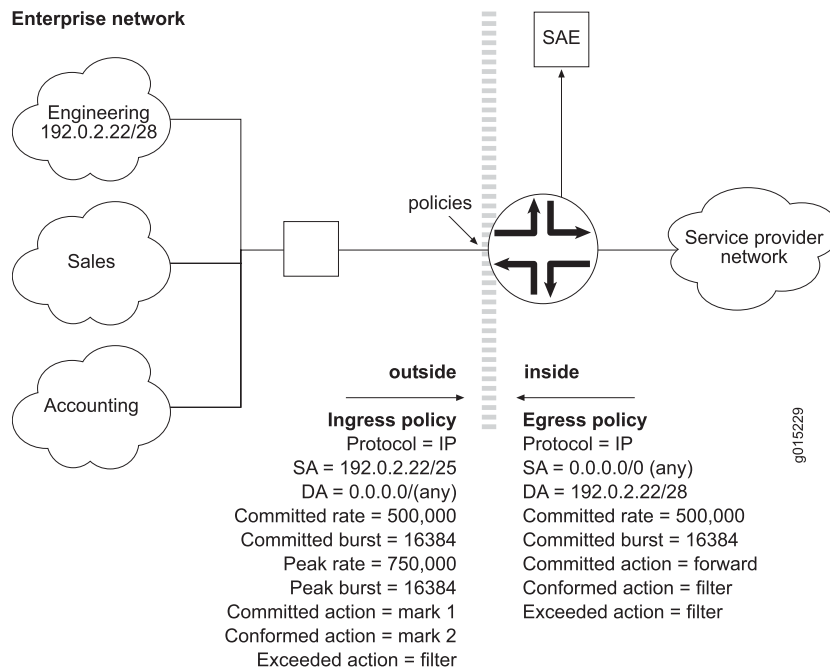
- outside = dept—Acquired from the service definition
- dept = eng—Acquired from the subscription
- eng = 192.0.2.22/28—Acquired from the enterprise subscriber definition
- qos = 500,000

This value was acquired from the service definition where the value of qos was set to 50 % of the interface\_speed parameter. An interface\_speed value of 1,000,000 was acquired from the router. If qos = 50 % of the interface speed, then the qos value is 500,000.

The rest of the rate-limit values are calculated based on the 500,000 value of qos.

Figure 2 on page 4 shows the values of the ingress and egress policies that are applied to the router in our sample network.

**Figure 2: Policies Applied to the Sample Network**



## Configuration

Configure a policy, service, subscriber, and subscription to use parameter value acquisition:

- Configuring the Default Value for a Global Parameter on page 4
- Configuring a Policy Group on page 5
- Configuring a Service on page 12
- Creating an Enterprise Subscriber on page 14
- Subscribing ABCInc to the GoldMetered Service on page 16

### Configuring the Default Value for a Global Parameter

Configure the global parameter any which is used in the policy configuration.

#### CLI Quick Configuration

To quickly configure the global parameter any, copy the following commands into a text editor, and modify them as needed; then load the configuration from the file.

```
[edit]
set policies global-parameters any default-value 0.0.0.0/0
set policies global-parameters any type network
```

#### Step-by-Step Procedure

To configure the global parameter any:

1. From configuration mode, enter the global parameter configuration for the any parameter.

```
[edit]
```

```
user@host# edit policies global-parameters any
```

2. (Optional) Configure a default value that the policy engine uses if no other values are provided during the parameter value acquisition process.

See Parameter Types for valid values of each parameter type.

```
[edit policies global-parameters any]
user@host# set default-value 0.0.0.0/0
```

3. (Optional) Type of attribute for which you can use the parameter.

```
[edit policies global-parameters any]
user@host# set type network
```

## Configuring a Policy Group

Configure the policy group tierpolicy to specify bandwidth for incoming and outgoing traffic.

### CLI Quick Configuration

To quickly configure the global parameter any, copy the following commands into a text editor, and modify them as needed; then load the configuration from the file.

```
[edit]
set policies folder ent group tierpolicy
set policies folder ent group tierpolicy local-parameters qos
set policies folder ent group tierpolicy local-parameters qos type rate

set policies folder ent group tierpolicy local-parameters outside
set policies folder ent group tierpolicy local-parameters outside type network
set policies folder ent group tierpolicy local-parameters outside default-value any
set policies folder ent group tierpolicy local-parameters inside
set policies folder ent group tierpolicy local-parameters inside type network
set policies folder ent group tierpolicy local-parameters inside default-value any

set policies folder ent group tierpolicy list egrules
set policies folder ent group tierpolicy list egrules role junose-ipv4
set policies folder ent group tierpolicy list egrules applicability output

set policies folder ent group tierpolicy list ingrules
set policies folder ent group tierpolicy list ingrules role junose-ipv4
set policies folder ent group tierpolicy list ingrules applicability input

set policies folder ent group tierpolicy list egrules rule eglimit
set policies folder ent group tierpolicy list egrules rule eglimit type junose-ipv4
set policies folder ent group tierpolicy list egrules rule eglimit precedence 1000
set policies folder ent group tierpolicy list egrules rule eglimit accounting

set policies folder ent group tierpolicy list egrules rule eglimit traffic-condition cond
set policies folder ent group tierpolicy list egrules rule eglimit traffic-condition cond
source-network group-network network-specifier inside
```

```
set policies folder ent group tierpolicy list egrules rule eglimit traffic-condition cond
destination-network group-network network-specifier outside
```

```
set policies folder ent group tierpolicy rate-limit ratelimit
set policies folder ent group tierpolicy type two-rate
```

```
set policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit
committed-rate qos
set policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit
committed-burst "max(qos*0.1, 16384)"
set policies folder ent group tierpolicy rate-limit ratelimit committed-action forward
set policies folder ent group tierpolicy rate-limit ratelimit exceed-action filter
set policies folder ent group tierpolicy rate-limit ratelimit conformed-action filter
set policies folder ent group tierpolicy rate-limit ratelimit exceed-action filter
```

```
set policies folder ent group tierpolicy list ingrules rule inglimit
set policies folder ent group tierpolicy list ingrules rule inglimit type junose-ipv4
set policies folder ent group tierpolicy list ingrules rule inglimit precedence 1000
set policies folder ent group tierpolicy list ingrules rule inglimit accounting
```

```
set policies folder ent group tierpolicy list ingrules rule inglimit traffic-condition ent
set policies folder ent group tierpolicy list ingrules rule inglimit traffic-condition ent
source-network group-network network-specifier outside
set policies folder ent group tierpolicy list ingrules rule inglimit traffic-condition ent
destination-network group-network network-specifier inside
```

```
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
type two-rate
```

```
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
committed-rate qos
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
committed-burst "max(qos*0.1, 16384)"
```

```
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
peak-rate qos*1.5
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
committed-action mark mark-info value 1
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
conformed-action mark mark-info value 2
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
exceed-action filter
set policies folder ent group tierpolicy list ingrules rule inglimit rate-limit rateLimit
peak-burst "max(qos*1.5*0.1, 16384)"
```

**Step-by-Step Procedure** To create and configure a policy group named tierpolicy:

1. Create the tiergroup policy.

```
[edit]
user@host# edit policies folder ent group tierpolicy
```

2. Create local parameters, which are parameters that will be used only with tierpolicy.

3. qos—Rate parameter

```
[edit policies folder ent group tierpolicy]
user@host# edit local-parameters qos
[edit policies folder ent group tierpolicy local-parameters qos]
user@host# set type rate
```

- outside—Network parameter with a default value of any; any is a global parameter with value 0.0.0.0/0, which matches any network

```
[edit policies folder ent group tierpolicy]
user@host# edit local-parameters outside
[edit policies folder ent group tierpolicy local-parameters outside]
user@host# set type network
[edit policies folder ent group tierpolicy local-parameters outside]
user@host# set default-value any
```

- inside—Network parameter with a default value of any; any is a global parameter with value 0.0.0.0/0, which matches any network

```
[edit policies folder ent group tierpolicy]
user@host# edit local-parameters inside
[edit policies folder ent group tierpolicy local-parameters inside]
user@host# set type network
[edit policies folder ent group tierpolicy local-parameters inside]
user@host# set default-value any
```

- Create a policy lists for egress side of the interface.

```
[edit policies folder ent group tierpolicy]
user@host# edit list egrules

[edit policies folder ent group tierpolicy list egrules]
user@host# set role junose-ipv4

[edit policies folder ent group tierpolicy list egrules]
user@host# set applicability output
```

- Create a policy list, for the ingress side of the interface.

```
[edit policies folder ent group tierpolicy]
user@host# edit list ingrules

[edit policies folder ent group tierpolicy list ingrules]
user@host# set role junose-ipv4

[edit policies folder ent group tierpolicy list ingrules]
user@host# set applicability input
```

- Create a policy rule for egress traffic.

```
[edit policies folder ent group tierpolicy list egrules]
user@host# edit rule eglimit
```

```
[edit policies folder ent group tierpolicy list egrules rule eglimit]
user@host# set type junose-ipv4
```

```
[edit policies folder ent group tierpolicy list egrules rule eglimit]
user@host# set precedence 1000
```

```
[edit policies folder ent group tierpolicy list egrules rule eglimit]
user@host# set accounting
```

- In the egress policy rule, which applies to traffic coming from the service provider network to the enterprise, create a condition that matches IP packets on source and destination networks:
- source network = inside
- destination network = outside

```
[edit policies folder ent group tierpolicy list egrules rule eglimit]
user@host# edit traffic-condition cond
```

```
[edit policies folder ent group tierpolicy list egrules rule eglimit traffic-condition
cond ]
user@host# set source-network group-network network-specifier inside
```

```
[edit policies folder ent group tierpolicy list egrules rule eglimit traffic-condition
cond ]
user@host# set destination-network group-network network-specifier outside
```

- Also in the egress policy rule, create a rate-limit action and set the type to the runtime parameter two-rate.

```
[edit policies folder ent group tierpolicy list egrules rule eglimit]
user@host# edit rate-limit ratelimit
```

```
[edit policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit]
user@host# set type two-rate
```

- Configure the rate-limit action in the egress policy rule to do the following:
- Set the committed rate to the qos parameter.

```
[edit policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit]
user@host# set committed-rate qos
```

- Set the committed burst to the maximum of either 800 ms burst at committed rate in bytes ( $\text{qos} \times 0.1$ ) or 16384.

```
[edit policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit]
user@host# set committed-burst "max(qos*0.1, 16384)"
```



- Use the default peak burst rate of 16384.
- Forward all committed traffic.
 

```
[edit policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit]
user@host# set committed-action forward
```
- Filter all uncommitted traffic.
 

```
[edit policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit]
user@host# set exceed-action filter
[edit policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit]
user@host# set conformed-action filter
[edit policies folder ent group tierpolicy list egrules rule eglimit rate-limit ratelimit]
user@host# set exceed-action filter
```
- Create a policy rule for ingress traffic.
 

```
[edit policies folder ent group tierpolicy list ingrules]
user@host# edit rule inglimit
```

```
[edit policies folder ent group tierpolicy list ingrules rule inglimit]
user@host# set type junose-ipv4
```

```
[edit policies folder ent group tierpolicy list ingrules rule inglimit]
user@host# set precedence 1000
```

```
[edit policies folder ent group tierpolicy list ingrules rule inglimit]
user@host# set accounting
```
- In the ingress policy rule, which applies to traffic coming from the enterprise network, create a condition that matches IP packets on source and destination networks:
  - source network = outside
  - destination network = inside

```
[edit policies folder ent group tierpolicy list ingrules rule inglimit]
user@host# edit traffic-condition ent
```

```
[edit policies folder ent group tierpolicy list ingrules rule inglimit traffic-condition ent]
user@host# set source-network group-network network-specifier outside
```

```
[edit policies folder ent group tierpolicy list ingrules rule inglimit traffic-condition ent]
user@host# set destination-network group-network network-specifier inside
```
- Also in the ingress policy rule, create a rate-limit action and set the type to the runtime parameter two-rate.
 

```
[edit policies folder ent group tierpolicy list ingrules rule inglimit]
```

```
user@host# edit rate-limit rateLimit
```

```
[edit policies folder ent group tierpolicy list ingrulers rule inglimit rate-limit rateLimit]  
user@host# set type two-rate
```

- Configure the rate-limit action in the ingress policy rule to do the following:
- Set the committed rate to the qos local parameter.

```
[edit policies folder ent group tierpolicy list ingrulers rule inglimit rate-limit rateLimit]  
user@host# set committed-rate qos
```

- Set the committed burst to either 800 ms burst or at the committed rate in bytes ( $\text{qos} \times 0.1$ ) or 16384.

```
[edit policies folder ent group tierpolicy list ingrulers rule inglimit rate-limit rateLimit]  
user@host# set committed-burst "max(qos*0.1, 16384)"
```

- Scale the peak rate and burst by 1.5.

```
[edit policies folder ent group tierpolicy list ingrulers rule inglimit rate-limit rateLimit]  
user@host# set peak-rate qos*1.5
```

- Mark committed and conformed traffic with different marks (1 and 2).

```
[edit policies folder ent group tierpolicy list ingrulers rule inglimit rate-limit rateLimit]  
user@host# set committed-action mark mark-info value 1  
[edit policies folder ent group tierpolicy list ingrulers rule inglimit rate-limit rateLimit]  
user@host# set conformed-action mark mark-info value 2
```

- Drop all traffic that exceeds the rate limit.

```
[edit policies folder ent group tierpolicy list ingrulers rule inglimit rate-limit rateLimit]  
user@host# set exceed-action filter
```

- Set the peak burst rate to the maximum of either 800 ms burst of one and a half times the committed rate in bytes ( $\text{qos} \times 1.5$ ) or 16384.

```
[edit policies folder ent group tierpolicy list ingrulers rule inglimit rate-limit rateLimit]  
user@host# set peak-burst "max(qos*1.5*0.1, 16384)"
```

## Configuration Results

```
[edit policies folder ent group tierpolicy]  
user@host# show  
description "This is a service policy for services that rate limit and account  
for traffic to and from the service provider's network. It is parameterized on  
the subnets inside and outside the service provider's network between which the  
traffic flows. It is also parameterized on a number which is used to scale ingress  
and egress rate limit rules. ";  
local-parameters {  
  qos {  
    description "Scaling factor to apply to the rate limits on the traffic between  
inside and outside";  
    type rate;
```

```

}
outside {
    description "the subnet outside the service provider's network";
    default-value any;
    type network;
}
inside {
    description "the subnet inside the service provider's network";
    default-value any;
    type network;
}
}
list egrules {
    role junose-ipv4;
    applicability output;
    rule eglimit {
        type junose-ipv4;
        precedence 1000;
        accounting;
        rate-limit ratelimit {
            committed-action {
                forward {
                }
            }
            conformed-action {
                filter {
                }
            }
            exceed-action {
                filter {
                }
            }
        }
        type two_rate;
        committed-rate qos;
        committed-burst "max(qos*0.1, 16384)";
        peak-rate qos*1.5;
        peak-burst 16384;
        description "committed rate is \"qos\" parameter, burst is 800ms burst at
        committed rate (*0.1 remember rates are bits per second, bursts are bytes)
        drop all uncommitted traffic. Max with 16384 to make sure burst is not too small
        for slow interfaces.
        ";
    }
    traffic-condition cond {
        source-network {
            group-network {
                network-specifier inside;
            }
        }
        destination-network {
            group-network {
                network-specifier outside;
            }
        }
    }
    description "rule to limit egress traffic";
}
}
list ingrules {
    role junose-ipv4;
    applicability input;

```

```

rule inglimit {
    type junose-ipv4;
    precedence 1000;
    accounting;
    rate-limit rateLimit {
        committed-action {
            mark {
                mark-info {
                    value 1;
                }
            }
        }
        conformed-action {
            mark {
                mark-info {
                    value 2;
                }
            }
        }
        exceed-action {
            filter {
            }
        }
    }
    type two_rate;
    committed-rate qos;
    committed-burst "max(qos*0.1, 16384)";
    peak-rate qos*1.5;
    peak-burst "max(qos*1.5*0.1, 16384)";
    description "committed rate is \"qos\" parameter, burst is 800ms burst at
    committed rate (*0.1 remember rates are bits per second, bursts are bytes). Max
    with 16384 to make sure burst is not too small for slow interfaces.peak rate and
    burst are scaled by 1.5. mark committed and conformed traffic with different
    marks, drop all excess traffic
    ";
}

traffic-condition ent {
    source-network {
        group-network {
            network-specifier outside;
        }
    }
    destination-network {
        group-network {
            network-specifier inside;
        }
    }
}
description "rule to limit ingress traffic";
}
}

```

## Configuring a Service

Configure a service that provides a gold-level quality of service to subscribers.

**CLI Quick Configuration** To quickly configure a service copy the following commands into a text editor, and modify them as needed; then load the configuration from the file.

```

[edit]
set services

```

```

set services scope EntJunose
set services scope EntJunose service GoldMetered
set services scope EntJunose service GoldMetered type normal
set services scope EntJunose service GoldMetered category "Quality of Service"
set services scope EntJunose service GoldMetered policy-group /ent/tierpolicy
set services scope EntJunose service GoldMetered radius-class GoldMetered
set services scope EntJunose service GoldMetered parameter substitution
[ "dept:network//the subnet of the department to apply the service to"
"!inside:network = any//always apply to any subnet inside the service provider"
"!outside:network = dept//rename outside policy parameter to dept" "!qos =
interface_speed*0.5//gold qos is 50% of interface speed" ]

```

**Step-by-Step Procedure** To configure a service that uses the policy tierpolicy:

1. Create a service called GoldMetered, and assign tierpolicy as the policy group.

```

[edit]
user@host# edit services

```

```

[edit services]
user@host# edit scope EntJunose

```

```

[edit services scope EntJunose]
user@host# edit service GoldMetered

```

```

[edit services scope EntJunose service GoldMetered]
user@host# set type normal

```

```

[edit services scope EntJunose service GoldMetered]
user@host# set category "Quality of Service"

```

```

[edit services scope EntJunose service GoldMetered]
user@host# set policy-group /ent/tierpolicy

```

```

[edit services scope EntJunose service GoldMetered]
user@host# set radius-class GoldMetered

```

2. Edit the parameter for the GoldMetered service, and add the following substitutions:

- dept—Create a parameter called dept that is parameter type (role) network. This is the subnet of the department that the service will apply to.
- qos—Fix the qos parameter to 50 % of the interface\_speed parameter. (interface\_speed is a global runtime parameter that the SAE fills in with the actual speed of the router interface).
- outside—Set the outside parameter to the value dept, which effectively renames the outside parameter to dept.
- inside—Set the inside parameter to a value of any, which applies to any subnet inside the service provider's network.

```
[edit services scope EntJunose service GoldMetered]
user@host# set parameter substitution [ "dept:network//the subnet of the
department to apply the service to" "!inside:network = any//always apply
to any subnet inside the service provider" "!outside:network = dept//rename
outside policy parameter to dept" "!qos = interface_speed*0.5//gold qos is
50% of interface speed" ]
```

## Configuration Results

```
[edit services scope EntJunose service GoldMetered]
user@host# show
description "Provides gold level quality of service to given enterprise department
subnet charged on volume";
type normal;
category "Quality of Service";
policy-group /ent/tierpolicy;
radius-class GoldMetered;
status active;
parameter {
  substitution [ "dept:network//the subnet of the department to apply the servic
e to" "!inside:network = any//always apply to any subnet inside the service prov
ider" "!outside:network = dept//rename outside policy parameter to dept" "!qos =
interface_speed*0.5//gold qos is 50% of interface speed" ];
}
```

## Creating an Enterprise Subscriber

Create the eng parameter for use in parameter substitution. this parameter represents an enterprise subscriber. You can configure the substitution in the SRC CLI, the sample enterprise service portal, or the C-Web interface.

## CLI Quick Configuration

To quickly configure the global parameter any, copy the following commands into a text editor, and modify them as needed; then load the configuration from the file.

```
[edit]
set subscribers retailer default subscriber-folder local enterprise ABCInc substitution
[ " acct : network = 208.93.36.80 / 28" "eng : network = 208.93.36.64 / 28"
]
set subscribers retailer default subscriber-folder local enterprise ABCInc substitution
[ "acct : network = 208.93.36.80 / 28" "eng : network = 208.93.36.64 / 28"
]
```

## Step-by-Step Procedure

To create a parameter called eng in an existing enterprise:

1. Create the eng parameter with parameter type (role) network, and set the value of eng to 192.0.2.22/28.

```
[edit subscribers retailer default subscriber-folder local enterprise ABCInc]
user@host# set substitution [ "acct : network = 208.93.36.80 / 28" "eng :
network = 208.93.36.64 / 28" ]
```

2. Create the eng parameter as part of the subscriber definition.

- To create the eng parameter with the SRC CLI:

```
[edit subscribers retailer default subscriber-folder local enterprise ABCInc]
user@host# set substitution [ "acct : network = 208.93.36.80 / 28" "eng :
network = 208.93.36.64 / 28" ]
```

- To create the eng parameter in the sample enterprise service portal, select the **Departments** tab, add eng to the department field, and enter 192.0.2.22/28 as the network address of the department.



Virneo Enterprise Portal

[Log out](#)

Navigation

- ent-admin
  - default
    - local
      - ABCInc
        - Boca
          - Backup
          - Primary
          - Boston
          - Montreal
          - Ottawa
          - PrimaryAccess
          - Acme
  - retailer-one
  - retailer-two
  - SP
  - virtual-SP

[Refresh](#)

default ▸ local ▸ ABCInc ▸ Boca ▸ Primary ▸

Services Subscriptions Sessions **Departments** Managers

Department	Department network	Locked	
eng	192.0.2.22/28 <small>(From enterprise ABCInc)</small>	<input type="checkbox"/>	<a href="#" style="background-color: #ccc; padding: 2px 5px;">Apply</a> <a href="#" style="background-color: #ccc; padding: 2px 5px;">Delete</a> <a href="#" style="background-color: #ccc; padding: 2px 5px;">Reset</a>
acct	208.93.36.80/28 <small>(From enterprise ABCInc)</small>	<input type="checkbox"/>	<a href="#" style="background-color: #ccc; padding: 2px 5px;">Apply</a> <a href="#" style="background-color: #ccc; padding: 2px 5px;">Delete</a> <a href="#" style="background-color: #ccc; padding: 2px 5px;">Reset</a>
<input style="width: 100%;" type="text"/>	<input style="width: 100%;" type="text"/>	<input type="checkbox"/>	<a href="#" style="background-color: #ccc; padding: 2px 10px;">Create</a>

© Juniper Networks 2003-2004

### Configuration Results

```
[edit subscribers retailer default subscriber-folder local enterprise ABCInc]
user@host# show
...
substitution [ "acct : network = 208.93.36.80 / 28" "eng : network = 208.93.36.64 / 28" ];
...
subscription GoldSecured {
    status active;
    activation manual;
    substitution "!dept : network = eng";
}
```

## Subscribing ABCInc to the GoldMetered Service

Subscribe to the GoldMetered service.

### Step-by-Step Procedure

To subscribe the ABCInc subscriber to the GoldMetered service through the sample enterprise service portal.

1. Select **ABCInc** in the navigation pane.
2. Select the **Services** tab.

The Services pane appears.



Virneo Enterprise Portal

Log out

Navigation

ent-admin

default

local

ABCInc

Boca

Boston

Montreal

Ottawa

PrimaryAccess

Acme

retailer-one

retailer-two

SP

virtual-SP

Refresh

default local ABCInc

Services Subscriptions Departments Managers

Service	Current local subscriptions	New local subscription name	
Internet-Gold		<input type="text"/>	Subscribe
News		<input type="text"/>	Subscribe
Video-Bronze		<input type="text"/>	Subscribe
Audio-Bronze		<input type="text"/>	Subscribe
PingDoSProtect	[unnamed]	<input type="text"/>	Subscribe
GoldMetered	[unnamed]	<input type="text"/>	Subscribe
GoldSecured		<input type="text"/>	Subscribe
BronzeMetered	[unnamed]	<input type="text"/>	Subscribe
Internet-Silver		<input type="text"/>	Subscribe

3. Click **Subscribe** in the GoldMetered service row.
4. Select the **Subscriptions** tab.

The Subscriptions pane appears.



Navigation

- ent-admin
  - default
    - local
      - ABCInc**
        - Boca
        - Boston
        - Montreal
        - Ottawa
        - PrimaryAccess
        - Acme
      - retailer-one
      - retailer-two
      - SP
      - virtual-SP

Refresh

default local **ABCInc**

Service	Subscription	Subscription details									
BronzeMetered	[unnamed]	<table border="1"> <thead> <tr> <th>Subscription Status</th> <th>Action</th> </tr> </thead> <tbody> <tr> <td>Administratively inactive.</td> <td> <input type="button" value="Activate"/> <input type="button" value="Deactivate"/> </td> </tr> <tr> <td>Not suspended.</td> <td> <input type="button" value="Unsuspend"/> <input type="button" value="Suspend"/> </td> </tr> <tr> <td>Usage</td> <td> <input type="button" value="Reporting"/> </td> </tr> </tbody> </table>	Subscription Status	Action	Administratively inactive.	<input type="button" value="Activate"/> <input type="button" value="Deactivate"/>	Not suspended.	<input type="button" value="Unsuspend"/> <input type="button" value="Suspend"/>	Usage	<input type="button" value="Reporting"/>	
Subscription Status	Action										
Administratively inactive.	<input type="button" value="Activate"/> <input type="button" value="Deactivate"/>										
Not suspended.	<input type="button" value="Unsuspend"/> <input type="button" value="Suspend"/>										
Usage	<input type="button" value="Reporting"/>										
GoldMetered	[unnamed]										
PingDoSPProtect	[unnamed]										

**Service Parameters**  
(use checkbox to lock value)

dept =  ☒

- In the dept = field of the Service Parameters box, set the value of the dept parameter to eng.

- Related Topics**
- Parameters and Substitutions
  - Value Acquisition for Single Subscriptions
  - Formatting Substitutions
  - Adding Comments to Substitutions
  - Validating Substitutions
  - Adding a Normal Service (SRC CLI)

