

InCommon®



# InCommon Certificate Manager

## Domain Control Validation API

InCommon  
c/o Internet2  
1000 Oakbrook Drive, Suite 300  
Ann Arbor MI, 48104

## 1 Introduction

Any domain added to Incommon CM must pass Domain Control Validation (DCV) before Incommon can issue certificates to it. In addition to completing DCV via the Incommon CM interface, Incommon CM also allows customers to complete DCV using our RESTful Application API.

Resource root:

"<Comodo CM\_root>/api/dcv/v1/validation". Generally this is a similar to operations accessible via GUI.

## 2 Start DCV

There are four possible methods of starting Domain Control Validation:

- [HTTP](#)
- [HTTPS](#)
- [CNAME](#)
- [Email](#)

Each method is described with JSON and XML format examples.

### 2.1 HTTP Method

**JSON example:**

HTTP POST /api/dcv/v1/validation/start/domain/http

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{
  "domain": "yourdomainname.com"
}
```

\*HTTP #status# 200 – OK, otherwise error

response:

```
{
  "url": "http://<yourdomainname>.com/.well-known/pki-validation/7aef859e3f5be9cea18785b0fb68e821.txt",
  "content": "be52a4441d1ea357e591beb6d49291ccda3de387c5fa24d66be74a146f0d7e6fncomodoca.com"
}
```

**XML example:**

HTTP POST /api/dcv/v1/validation/start/domain/http

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>
  <domain> yourdomainname.com </domain>
</domainRequest>
```

response:

```
<domainHttpResponse>
  <url>
    http://yourdomainname.com/.well-known/pki-validation/7aef859e3f5be9cea18785b0fb68e821.txt
  </url>
<content>
be52a4441d1ea357e591beb6d49291ccda3de387c5fa24d66be74a146f0d7e6fComodoca.com
</content>
</domainHttpResponse>
```

## 2.2 HTTPS Method

**JSON example:**

HTTP POST /api/dcv/v1/validation/start/domain/https

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{
  "domain": "yourdomainname.com"
}
```

\*HTTP #status# 200 – OK, otherwise error

response:

```
{
  "url": "https://yourdomainname.com/.well-known/pki-validation/7aef859e3f5be9cea18785b0fb68e821.txt ",
  "content": "be52a4441d1ea357e591beb6d49291ccda3de387c5fa24d66be74a146f0d7e6fncomodoca.com"
}
```

**XML example:**

HTTP POST /api/dcv/v1/validation/start/domain/https

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>
  <domain> yourdomainname.com </domain>
</domainRequest>
```

response:

```
<domainHttpsResponse>
  <url>
https://yourdomainname.com/.well-known/pki-validation/7aef859e3f5be9cea18785b0fb68e821.txt
  </url>
  <content>
be52a4441d1ea357e591beb6d49291ccda3de387c5fa24d66be74a146f0d7e6fComodoca.com
  </content>
</domainHttpsResponse>
```

## 2.3 CNAME Method

**JSON example:**

HTTP POST /api/dcv/v1/validation/start/domain/cname

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{
  "domain": "yourdomainname.com"
}
```

\*HTTP #status# 200 – OK, otherwise error

response:

```
{
  "host": "_5f3ab9e0619423ec8e3964a59a2eb0b8.yourdomainname.com. ",
  "point": "0e709ea2d503f546be25fabb46b19d33.b8623e4f94edf161911e2a829e17e214.comodoca.com. "
}
```

**XML example:**

HTTP POST /api/dcv/v1/validation/start/domain/cname

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>
  <domain> yourdomainname.com </domain>
</domainRequest>
```

response:

```
<domainCnameResponse>
  <host> _5f3ab9e0619423ec8e3964a59a2eb0b8.yourdomainname.com. </host>
<point> 0e709ea2d503f546be25fabb46b19d33.b8623e4f94edf161911e2a829e17e214.comodoca.com.
</point>
</domainCnameResponse>
```

## 2.4 Email Method

**JSON example:**

HTTP POST /api/dcv/v1/validation/start/domain/email

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{
  "domain": "yourdomainname.com"
}
```

\*HTTP #status# 200 – OK, otherwise error

response:

```
{
  "emails": [
    "admin@yourdomainname.com",
    "administrator@yourdomainname .com",
    "hostmaster@yourdomainname .com",
    "postmaster@yourdomainname .com",
```

```
"webmaster@yourdomainname .com"
```

```
]
```

```
}
```

**XML example:**

HTTP POST /api/dcv/v1/validation/start/domain/email

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>
```

```
    <domain> yourdomainname.com </domain>
```

```
</domainRequest>
```

response:

```
<domainEmailResponse>
```

```
    <emails> admin@yourdomainname.com </emails>
```

```
    <emails> administrator@yourdomainname .com </emails>
```

```
    <emails> hostmaster@yourdomainname .com </emails>
```

```
    <emails> postmaster@yourdomainname .com </emails>
```

```
    <emails> webmaster@yourdomainname .com </emails>
```

```
</domainEmailResponse>
```

## 3 Submit DCV

The following sections describe DCV submission using the RESTful API:

- [HTTP](#)
- [HTTPS](#)
- [CNAME](#)
- [Email](#)

Each of the method is described with JSON and XML format examples.

### 3.1 HTTP Method

**JSON example:**

HTTP POST /api/dcv/v1/validation/submit/domain/http

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{  
    "domain": "yourdomainname.com"  
}
```

\*HTTP #status# 200 – OK, otherwise error

\*Fields description / possible values:

"status" - "NOT\_VALIDATED", "VALIDATED" and "EXPIRED".

"orderStatus" - "NOT\_INITIATED", "AWAITING\_SUBMISSION" and "SUBMITTED".

response:

```
{  
    "status": "VALIDATED",  
    "orderStatus": "SUBMITTED",  
    "message": "DCV status: Validated DCV order status: Submitted"  
}
```

**XML example:**

HTTP POST /api/dcv/v1/validation/submit/domain/http

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>  
    <domain> yourdomainname.com </domain>  
</domainRequest>
```

response:

```
<submitDomainResponse>  
    <status> VALIDATED </status>  
    <orderStatus> SUBMITTED </orderStatus>  
    <message> DCV status: Validated DCV order status: Submitted </message>  
</submitDomainResponse>
```

## 3.2 HTTPS Method

**JSON example:**

HTTP POST /api/dcv/v1/validation/submit/domain/https

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{  
  "domain": "yourdomainname.com"  
}
```

\*HTTP #status# 200 – OK, otherwise error

\*Fields description / possible values:

"status" - "NOT\_VALIDATED", "VALIDATED" and "EXPIRED".

"orderStatus" - "NOT\_INITIATED", "AWAITING\_SUBMISSION" and "SUBMITTED".

response:

```
{  
  "status": "VALIDATED",  
  "orderStatus": "SUBMITTED",  
  "message": "DCV status: Validated DCV order status: Submitted"  
}
```

#### XML example:

HTTP POST /api/dcv/v1/validation/submit/domain/https

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>  
  <domain> yourdomainname.com </domain>  
</domainRequest>
```

response:

```
<submitDomainResponse>  
  <status> VALIDATED </status>  
  <orderStatus> SUBMITTED </orderStatus>  
  <message> DCV status: Validated DCV order status: Submitted </message>  
</submitDomainResponse>
```



### 3.3 CNAME Method

#### JSON example:

HTTP POST /api/dcv/v1/validation/submit/domain/cname

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{  
  "domain": "yourdomainname.com"  
}
```

\*HTTP #status# 200 – OK, otherwise error

\*Fields description / possible values:

*"status"* - "NOT\_VALIDATED", "VALIDATED" and "EXPIRED".

*"orderStatus"* - "NOT\_INITIATED", "AWAITING\_SUBMISSION" and "SUBMITTED".

response:

```
{  
  "status": "VALIDATED",  
  "orderStatus": "SUBMITTED",  
  "message": "DCV status: Validated DCV order status: Submitted"  
}
```

#### XML example:

HTTP POST /api/dcv/v1/validation/submit/domain/cname

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>  
  <domain> yourdomainname.com </domain>  
</domainRequest>
```

response:

```
<submitDomainResponse>  
  <status> VALIDATED </status>
```

```
<orderStatus> SUBMITTED </orderStatus>
```

```
<message> DCV status: Validated DCV order status: Submitted </message>
```

```
</submitDomainResponse>
```

### 3.4 Email Method

#### JSON example:

HTTP POST /api/dcv/v1/validation/submit/domain/email

HTTP Header:

Content-Type: application/json

Accept: application/json

#### request:

```
{  
  "domain": "yourdomainname.com",  
  "email": "admin@yourdomainname.com"  
}
```

\*HTTP #status# 200 – OK, otherwise error

\*Fields description / possible values:

"status" - "NOT\_VALIDATED", "VALIDATED" and "EXPIRED".

"orderStatus" - "NOT\_INITIATED", "AWAITING\_SUBMISSION" and "SUBMITTED".

#### response:

```
{  
  "status": "VALIDATED",  
  "orderStatus": "SUBMITTED",  
  "message": "DCV status: Validated DCV order status: Submitted"  
}
```

#### XML example:

HTTP POST /api/dcv/v1/validation/submit/domain/email

HTTP Header:

Content-Type: application/xml

Accept: application/xml

#### request:

```
<domainEmailRequest>  
  <domain> yourdomainname.com </domain>  
  <email> admin@yourdomainname.com </email>
```

```
</domainEmailRequest>
```

response:

```
<submitDomainResponse>
```

```
  <status> VALIDATED </status>
```

```
  <orderStatus> SUBMITTED </orderStatus>
```

```
  <message> DCV status: Validated DCV order status: Submitted </message>
```

```
</submitDomainResponse>
```

## 4 DCV Status

### JSON example:

HTTP POST /api/dcv/v1/validation/status

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{
  "domain": "yourdomainname.com"
}
```

\*HTTP #status# 200 – OK, otherwise error

\*Fields description / possible values:

"status" - "NOT\_VALIDATED", "VALIDATED" and "EXPIRED".

response:

```
{
  "status": "VALIDATED"
}
```

### XML example:

HTTP POST /api/dcv/v1/validation/status

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>
```

```
  <domain> yourdomainname.com </domain>
```

```
</domainRequest>
```

response:

```
<dcvStatusResponse>
```

```
    <status> VALIDATED </status>
```

```
</dcvStatusResponse>
```

## 5 Reset DCV

### JSON example:

HTTP POST /api/dcv/v1/validation/clear

HTTP Header:

Content-Type: application/json

Accept: application/json

request:

```
{
    "domain": "yourdomainname.com"
}
```

\*HTTP #status# 200 – OK, otherwise error

\*Fields description / possible values:

"status" - "NOT\_VALIDATED", "VALIDATED" and "EXPIRED".

"orderStatus" - "NOT\_INITIATED", "AWAITING\_SUBMISSION" and "SUBMITTED".

response:

```
{
    "status": "NOT_VALIDATED",
    "orderStatus": "NOT_INITIATED",
    "message": "DCV status: Not Validated DCV order status: Not Initiated"
}
```

### XML example:

HTTP POST /api/dcv/v1/validation/clear

HTTP Header:

Content-Type: application/xml

Accept: application/xml

request:

```
<domainRequest>
```

```
  <domain> yourdomainname.com </domain>
```

```
</domainRequest>
```

response:

```
<dcvCleanResponse>
```

```
  <status> NOT_VALIDATED </status>
```

```
  <orderStatus> NOT_INITIATED </orderStatus>
```

```
  <message> DCV status: Not Validated DCV order status: Not Initiated </message>
```

```
</dcvCleanResponse>
```