SPF Record Syntax

We have developed this comprehensive guide to increase your SPF understanding and help troubleshoot issues our application might have brought to your attention. Having a valid, accurate, and aligned SPF record will lead to improved authentication coverage, deliverability and help promote your desired level of security for your domains.

Don't have a dmarcian account? You can still query the contents of your SPF record using our SPF Survey tool.

Create a free account now to have dmarcian monitor your SPF, DKIM and DMARC records for you automatically. Get instant visibility into delivery errors, phishing and impersonation attempts with dmarcian's SaaS Platform.

Use the navigation menu just below to jump to the particular element of your SPF record in question. Additional information about SPF can be found in the linked articles at the bottom of this document. If you are new to SPF, we show you how to How to Create and Add an SPF Record.

Mechanisms: *all* *ip4* *ip6* *a* *mx* *ptr* *exists* *include* **Modifiers:** *redirect* *exp* *Too many lookups?*

can be prefixed with one of four qualifiers:

- \+ (Pass)
- – (Fail)
- ~ (SoftFail)
- ? (Neutral)

If a mechanism results in a hit, its qualifier value is used. The default qualifier is "+", i.e. "Pass". Mechanisms are evaluated in order. If no mechanism or modifier matches, the default result is "Neutral".

More in-depth information on the differences between "~" and "–" can be found here

Examples:

"v=spf1 -all"

"v=spf1 a -all"

"v=spf1 a mx -all"

"v=spf1 +a +mx -all"

If a domain has no SPF record at all, the result is "None". If a domain has a temporary error during DNS processing, you get the result "TempError" (called "error" in earlier drafts). If a syntax or evaluation error occurs (eg. the domain specifies an unrecognized mechanism) the result is "PermError" (formerly "unknown"). Evaluation of an SPF record can return any of these results:

Result	*Explanation*	*Intended action*
Pass	The SPF record designates the host to be allowed to send	accept

Fail	The SPF record has designated the host as NOT being allowed to send	reject
SoftFail	The SPF record has designated the host as NOT being allowed to send but is in transition	accept but mark
Neutral	The SPF record specifies explicitly that nothing can be said about validity	accept
None	The domain does not have an SPF record or the SPF record does not evaluate to a result	accept
PermError	A permanent error has occurred (eg. badly formatted SPF record)	unspecified
TempError	A transient error has occurred	accept or reject

The "all" mechanism

```
all
```

This mechanism always matches. It should always go at the end of the SPF record.

Examples:

“v=spf1 mx ~all”
Allow domain’s MXs to send mail for the domain, prohibit all others.

“v=spf1 -all”
The domain sends no mail at all.

“v=spf1 +all”
The domain allows all IP addresses on the internet to send mail. Though ‘valid’, this is not recommended.

The "ip4" mechanism

```
ip4:<ip4-address>
ip4:<ip4-network>/<prefix-length>
```

The argument to the “ip4:” mechanism is an IPv4 network range. If no p*refix-length* is given, /32 is assumed (singling out an individual host address). Be careful to include a prefix-length greater than /16, as delivery to some smaller receivers may be impacted.

Examples:

“v=spf1 ip4:192.168.0.1/16 ~all”
Allow any IP address between 192.168.0.1 and 192.168.255.255.

The "ip6" mechanism

```
ip6:<ip6-address>
ip6:<ip6-network>/<prefix-length>
```

The argument to the “ip6:” mechanism is an IPv6 network range. If no *prefix-length* is given, /128 is assumed (singling out an individual host address).

Examples:

“v=spf1 ip6:1080::8:800:200C:417A/96 ~all”
Allow any IPv6 address between 1080::8:800:0000:0000 and 1080::8:800:FFFF:FFFF.

“v=spf1 ip6:1080::8:800:68.0.3.1/96 ~all”
Allow any IPv6 address between 1080::8:800:0000:0000 and 1080::8:800:FFFF:FFFF.

The "a" mechanism

```
a
a/<prefix-length>
a:<domain>
a:<domain>/<prefix-length>
```

All the A records for *domain* are tested. If the client IP is found among them, this mechanism matches. If the connection is made over IPv6, then an AAAA lookup is performed instead.

If *domain* is not specified, the *current domain* is used.

The A records have to match the client IP exactly, unless a *prefix-length* is provided, in which case each IP address returned by the A lookup will be expanded to its corresponding CIDR prefix, and the client IP will be sought within that subnet.

Examples:

“v=spf1 a ~all”
The current domain is used.

“v=spf1 a:example.com ~all”
Equivalent if the current domain is example.com.

“v=spf1 a:mailers.example.com ~all”
Perhaps example.com has chosen to explicitly list all the outbound mailers in a special A record under mailers.example.com.

“v=spf1 a/24 a:offsite.example.com/24 ~all”
If example.com resolves to 192.0.2.1, the entire class C of 192.0.2.0/24 would be searched for the client IP. Similarly for offsite.example.com. If more than one A record were returned, each one would be expanded to a CIDR subnet.

The "mx" mechanism

```
mx
mx/<prefix-length>
mx:<domain>
mx:<domain>/<prefix-length>
```

All the A records for all the MX records for *domain* are tested in order of MX priority. If the client IP is found among them, this mechanism matches.

If *domain* is not specified, the *current domain* is used.

The A records have to match the client IP exactly, unless a prefix-length is provided, in which case each IP address returned by the A lookup will be expanded to its corresponding CIDR prefix, and the client IP will be sought within that subnet.

Examples:

“v=spf1 mx mx:deferrals.domain.com ~all”
Perhaps a domain sends mail through its MX servers plus another set of servers whose job is to retry mail for deferring domains.

“v=spf1 mx/24 mx:offsite.domain.com/24 ~all”
Perhaps a domain’s MX servers receive mail on one IP address, but send mail on a different but nearby IP address.

The "ptr" mechanism

```
ptr
ptr:<domain>
```

The hostname or hostnames for the client IP are looked up using PTR queries. The hostnames are then validated: at least one of the A records for a PTR hostname must match the original client IP. Invalid hostnames are discarded. If a valid hostname ends in domain, this mechanism matches.

If domain is not specified, the current domain is used.

If possible, you should avoid using this mechanism in your SPF record, because it will result in a larger number of expensive DNS lookups.

Examples:

“v=spf1 ptr ~all”
A domain which directly controls all its machines (unlike a dialup or broadband ISP) allows all its servers to send mail. For example, hotmail.com or paypal.com might do this.

“v=spf1 ptr:otherdomain.com ~all”
Any server whose hostname ends in otherdomain.com is designated.

The "exists" mechanism

```
exists:<domain>
```

Perform an A query on the provided domain. If a result is found, this constitutes a match. It doesn’t matter what the lookup result is – it could be 127.0.0.2.

When you use macros with this mechanism, you can perform RBL-style reversed-IP lookups, or set up per-user exceptions.

Examples:

In the following example, the client IP is 1.2.3.4 and the current domain is example.com.

“v=spf1 exists:example.com ~all”

If example.com does not resolve, the result is fail. If it does resolve, this mechanism results in a match.

The "include" mechanism

include:<domain>

The specified *domain* is searched for a match. If the lookup does not return a match or an error, processing proceeds to the next directive. ***Warning****:* If the *domain* does not have a valid SPF record, the result is a permanent error. Some mail receivers will reject based on a *PermError*.

Examples:

In the following example, the client IP is 1.2.3.4 and the current domain is example.com.

“v=spf1 include:example.com ~all”

If example.com has no SPF record, the result is PermError.
Suppose example.com’s SPF record were “v=spf1 a ~all”.
Look up the A record for example.com. If it matches 1.2.3.4, return Pass.
If there is no match, other than the included domain’s “~all”, the include as a whole fails to match; the eventual result is still Fail from the outer directive set in this example

Trust relationships — The “include:” mechanism is meant to cross administrative boundaries. Great care is needed to ensure that “include:” mechanisms do not place domains at risk for giving SPF Pass results to messages that result from cross user forgery. Unless technical mechanisms are in place at the specified other domain to prevent cross user forgery, “include:” mechanisms should give a Neutral rather than Pass result. This is done by adding “?” in front of “include:”.

The example would then be:

“v=spf1 ?include:example.com ~all”

Modifiers

Modifiers are optional. A modifier may appear only once per record. Unknown modifiers are ignored.

The "redirect" modifier

redirect=<domain>

The SPF record for *domain* replaces the current record. The macro-expanded *domain* is also substituted for the *current domain* in those lookups.

If a ‘redirect’ modifier is used, the SPF record should not also include the ‘all’ mechanism. If both are present, the ‘redirect’ modifier is ignored. Any ‘redirect’ modifiers beyond the first will be ignored.

Examples:

In the following example, the client IP is 1.2.3.4 and the current domain is example.com.

"v=spf1 redirect=example.com"

If example.com has no SPF record, that is an error; the result is unknown.
Suppose example.com's SPF record was "v=spf1 a ~all".
Look up the A record for example.com. If it matches 1.2.3.4, return Pass.
If there is no match, the exec fails to match, and the ~all value is used.

The "exp" modifier

```
exp=<domain>
```

If an SMTP receiver rejects a message, it can include an explanation. An SPF publisher can specify the explanation string that senders see. This way, an ISP can direct nonconforming users to a web page that provides further instructions about how to configure SASL.

The domain is expanded; a TXT lookup is performed. The result of the TXT query is then macro-expanded and shown to the sender. Other macros can be used to provide a customized explanation.

The exp modifier may only contain printable ASCII characters.

Too many lookups?

Over the past decade, it has become increasingly easier to send email. Countless Sources have entered the marketplace, each providing a specialized toolset tailored to address modern day needs of marketers, developers, and small businesses. Along with this expansion, email authentication, specifically SPF, has become an increasingly complex matter to navigate.

Within the SPF RFC specification (essentially internet law) there lies a practical limit of how many "DNS-querying mechanisms" a single SPF record can contain. That limit is ten. The ten maximum lookup states that a domain administrator (that's you!) will not require the likes of Gmail or other receivers to conduct more than ten consecutive DNS lookups to see if you authorize a particular IP address to send mail on your behalf.

As it has become somewhat commonplace for any single organization to authorize a large number of disparate netblocks (due to the outsourced nature of email infrastructure), there remains what seems like the constant and unnecessary encroachment on the ten maximum lookup. *This limit however remains entirely practical and should be observed to ensure timely delivery and favorable inbox rates.* Further, the solution to avoid the limit is squarely addressed by other mainstream email best practices, long encouraged by major inbound receivers such as Gmail and Yahoo.

The single most practical solution to avoid the 'too many lookups' issue is to make use of sub-domains. As each discrete sub.domain is afforded its own ten lookup maximum, SPF is effectively boundless. *Example: hello.com is permitted ten lookups + sub.hello.com is permitted ten lookups.* Plainly put, you should never run in to the ten maximum lookup condition if you are correctly segmenting different mail streams (eg. transactional, corporate, marketing, etc.) on to discrete name space.

In this sub section "delivery tips' of the Gmail postmaster site, it is recommended to;

- Use separate email addresses
- Send mail from different domains and/or IP addresses

In summary, you should not run in to the 10 lookup maximum. If you do, we've outlined some additional strategies and knowledge-base materials on how to navigate.