All of Government

Secure Government Email DKIM Deployment Guide

Issued by

Digital Services branch

Internal Affairs
Te Tari Taiwhenua

New Zealand Government

Contents

1 Document Control

Service Name	Secure Government Email
Author	Matt Oliver, Operations Security Specialist, Department of Internal Affairs
Title	Secure Government Email DKIM Deployment Guide
Date and Version	4 November 2025, v1.0
Index Number	AoGSD.SGE.Guidance.2025_250

2 Introduction

The Secure Government Email (SGE) Common Implementation Framework was confirmed late in 2024 with a goal to improve email security standards across the New Zealand Government. This deployment guide complements the Framework through providing examples on how to deploy Domain Keys Identified Mail (DKIM) in line with both the framework and to meet the requirements of the New Zealand Information Security Manual (NZISM). The full framework can be found here: Secure Government Email | NZ Digital government.

This document provides configuration examples covering commonly used platforms and configurations but is not an exhaustive guide. It is recommended Agencies without experience in the deployment of email security tools engage with a provider with expertise in this area.
Services are available for Agencies via the Secure Email Management and Administration service in Marketplace Marketplace | Pae Hokohoko — Infrastructure Managed Services catalogue.

3 NZISM and SGE Framework requirements for DKIM

Chapter 15.2 (Email Infrastructure) of the NZISM requires DKIM be configured across all domains irrespective of whether those domains are used in association with email. It also requires inbound email be scanned and acted on with DKIM compliance checks.

The SGE Framework maps back to the NZISM and requires that:

- All outbound email from all sending services must be DKIM signed.
- Blank DKIM records must be configured on non-email enabled domains.
- Inbound emails must be scanned and acted on with DKIM compliance checks*.

*Scanning of inbound emails for DKIM compliance is a part of Domain-based Message Authentication, Reporting, and Conformance (DMARC) and is covered in the Secure Government Email DMARC Deployment Guide.

4 DKIM Configuration

Enabling DKIM requires configuration changes in the sending email service, and in your domains' DNS environment. Configuring DKIM is similar across most platforms, although the exact configuration steps may differ. You will:

- Configure a private key for signing emails.
- Configure DNS with the associated public key or a CNAME pointer to that key.
- Configure the mail sending service to cryptographically sign outbound emails.

DKIM signatures are cryptographically bound to the exact message content — even a tiny modification (like adding a disclaimer or rewriting a header) will break the signature. Therefore, it is important that DKIM signing happen at the final sending hop in your Mail Exchange (MX) flow. If you use an external email security provider to process your outbound email, DKIM signing must happen at that point. A single email can be signed multiple times by DKIM.

Examples in this document use the fictitious domain name minties.govt.nz appearing as if it were hosted on an M365 instance named testlab5 <testlab5.onmicrosoft.com>.

It is assumed you have the required access and knowledge to configure the required services.

4.1 Deploying DKIM with M365 Exchange Online and Govt DNS

Microsoft hosts the required DKIM keys securely within their own infrastructure. When setting up DKIM on M365, Microsoft will provide you with two CNAME records for DNS which point recipient device DKIM record lookups to their location. This allows M365 to publish public keys on your behalf and rotate and manage those keys automatically. This is basically a "set and forget" type of setup.

Step 1: Identify Your Domain you wish to enable DKIM signing on.

1. minties.govt.nz.

Step 2: Access Microsoft 365 Defender Portal

1. Go to https://security.microsoft.com.
2. Navigate to Email & Collaboration > Policies & Rules > Threat Policies.
3. Under Email Authentication Settings, select DKIM.
4. Select the domain you wish to enable DKIM on and click on Create DKIM Keys.

This will provide you with two CNAME records which you need to add to your DNS:

```
Host Name : selector1._domainkey

Points to address or value: selector1-minties-govt-nz._domainkey.testlab5.n-
v1.dkim.mail.microsoft

Host Name : selector2._domainkey

Points to address or value: selector2-minties-govt-nz._domainkey.testlab5.n-
v1.dkim.mail.microsoft
```

These are only examples, do not copy and paste them from this document as they will not work. When configuring the settings within M365 make sure you copy the M365 statements exactly as provided.

Step 3: Add CNAME Records to Your DNS

1. Log in to the DNS portal and browse to your domain.

2. Click on the Current tab under DNS Zones.
3. Click on Edit.
4. Click on the + symbol to add a new row.
5. Paste in the host name from the record created in M365.
6. Set the TTL to 3600. (or whatever is required for your domain)
7. Change the type to CNAME.
8. Paste the points to address or value from the record created in M365.
9. Add a comment or change number if needed for your environment.
10. Repeat steps 4-9 for the second CNAME record from M365.
11. Click on Publish.

Step 4: Enable DKIM Signing

Once DNS records are published:

1. Go back to the DKIM settings in the Defender portal.

2. Select your domain.

3. Under "Sign messages for this domain with DKIM signatures," move the slider from disabled to enabled.

Microsoft will now cryptographically sign outbound mail using DKIM.

What Happens Behind the Scenes with DKIM when an email is sent via M365?

1. Outbound mail from your M365 tenancy is cryptographically signed using your private DKIM key, which is stored securely in Microsoft's infrastructure.
2. The receiving mail system looks up the corresponding public DKIM key via DNS — at the CNAME location.
3. The CNAME record redirects the query to Microsoft's hosted DKIM record under your onmicrosoft.com domain.
4. Microsoft returns the public DKIM key, which the recipient then uses to verify the signature and confirm message integrity and authenticity.

4.2 Deploying DKIM when you have an external mail security service

This section covers the typical deployment steps when using an external mail security service in your outbound mail flow. The most common scenario is using M365 as the mail platform then having your outbound email route via an external mail security service or gateway. There are many providers, and exact steps will vary between providers. The following is a generic list of steps to follow:

Step 1: Identify Your Outbound Mail Domain(s)

- Before enabling DKIM, confirm which sending domains the gateway will handle. e.g. minties.govt.nz. Each domain will require its own DKIM setup (keys + DNS record(s)).

Step 2: Generate or Request DKIM Keys

- There are two models:

Model	Description	Who Generates Keys
Customer-Managed DKIM	You create the DKIM key pair and give the public key to the service.	You (using as tool like openssl or the service's tool)
Service Provider-Managed DKIM	The service generates and manages the keys for you.	The mail security provider

The use of 2048-bit RSA keys are strongly recommended.

You'll end up with:

- A selector name (e.g., selector1)
- A public key (the part that goes in DNS)
- A private key (securely stored by the service for signing)

Step 3: Publish the DKIM Public Key in DNS

- Create a TXT record under your sending domain.
  The host (record name) and value typically look like:

```
Host / Name:   selector1._domainkey.yourcompany.com
Value:         v=DKIM1; k=rsa; p=MIIBIjANBgkqhkiG9w0BAQEFA..(truncated).
```

Step 4: Enable DKIM Signing in the Security Gateway Portal

- In your mail security service's admin console, find the outbound policy or Authentication / DKIM settings, then:
  1. Add a new DKIM profile or domain entry.
  2. Enter the domain name and selector.
  3. Upload or link the private key (if customer-managed) or confirm DNS publishing (if service provider-managed).
  4. Enable signing for outbound messages.

  Optional settings might include:

  - Key length / rotation schedule.
  - Which headers to sign (e.g. From, Subject, Date, Message-ID).
  - Whether to sign internal-to-external only or all outbound.

Example mail flow when using M365 behind an external gateway

1. Microsoft 365 → outbound connector → External Gateway.
2. Gateway applies DKIM signature with d=yourcompany.com.
3. Gateway delivers message to Internet recipient.
4. Recipient server validates DKIM signature using your DNS record.
5. Recipient server delivers email to the end users mailbox. (if authenticated)

4.3 Deploying DKIM on external mail sending services

There are many bulk Email Service Providers (ESPs) and Customer Relationship Management (CRM) platforms which need to be configured for DKIM. All major ESPs support DKIM signing and crucially DKIM alignment for DMARC checks. Though configurations will vary across service providers, the basic flow typically requires:

1. Verifying your domain.
2. Setting up domain authentication / selectors.
3. Publishing the DKIM records in DNS.
4. Enabling DKIM signing.

Most providers will generate the DKIM keys for you, and you will not have access to the private keys.

The following is an example for setting up DKIM on the Mailchimp platform.

(This is not an endorsement of the Mailchimp platform, we were specifically asked to provide this example.)

Step 1. Verify Your Domain in Mailchimp

Log in to your Mailchimp account.

Navigate to Account Settings → Domains.

Click Add & Verify Domain.

Enter an email address at the domain you want to verify (e.g., you@yourdomain.com).

Mailchimp will send a verification email. Click the link in the email to complete verification.

Step 2. Start Domain Authentication

Once verified, click Start Authentication next to your domain.

Mailchimp will provide two CNAME records for DKIM:

```
Type: CNAME
Host: k2._domainkey.yourdomain.com
Value: dkim2.mcsv.net

Type: CNAME
Host: k3._domainkey.yourdomain.com
Value: dkim3.mcsv.net
```

These records must be added to your domain's DNS settings via your DNS provider (e.g., Govt DNS, Cloudflare, GoDaddy).

Step 3. Publish the DKIM Records

Log in to your DNS provider.

Add the two CNAME records exactly as provided.

Save the config and allow time for DNS propagation (it can take a while).

Step 4. Confirm DKIM Authentication

Return to Mailchimp and click Authenticate Domain.

Mailchimp will check the DNS records and confirm DKIM setup.

Once Mailchimp has confirmed the DKIM setup all outbound mail will be signed.

DKIM Key Management in Mailchimp

- When you authenticate your domain in Mailchimp, they generate a private/public key pair.
- The public key is shared with you in the form of CNAME records that you publish in your DNS.
- The private key remains securely stored on Mailchimp's servers and is used to sign outgoing emails from your authenticated domain.

This setup ensures:

- You don't have to manage or rotate keys manually.
- Mailchimp handles the cryptographic signing process.
- The DKIM signature uses Mailchimp's domain (mcsv.net) but aligns with your domain via CNAME delegation.

Why This Matters for Strict Alignment:

- Even though the DKIM signature is technically from mcsv.net, the CNAME delegation allows it to appear as if it's coming from yourdomain.com, satisfying DMARC strict alignment (adkim=s) as long as the d= domain in the DKIM signature matches the "From" domain exactly.

4.4 Deploying DKIM on non-mail-enabled domains

The NZISM requires blank DKIM records be configured on all non-email-enabled domains. This is to force any DKIM lookup to 'fail' rather than to just receive no response. It is achieved through using a wildcard selector and providing no public key.

As these domains are not enabled for email the following DKIM record will only have an impact on any spoofed email from the domain. It is designed to be used in conjunction with a blank SPF record.

Example DKIM Record:

Name	TTL	Type	Data	Comment
*._domainkey<yourdomainname>	3600	txt	"v=DKIM1; p="	DKIM record with no public key

4.5 Testing DKIM

Once DKIM signing is operational on the sending platform you can test it by sending an email to yourself. Once received check the headers for the existence of a DKIM signature. You should see something similar to the following:

```
Authentication-Results: mx.microsoft.com;
       dkim=pass header.i=@minties.govt.nz header.s=s1 header.b=rRHfhgt6;
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed; d=minties.govt.nz; h=content-
type:date:from:mime-version:subject:reply-to:list-unsubscribe:list-unsubscribe-
post:to:cc:content-type:date:feedback-id:from:subject:to; s=s1;
bh=PRMpynrsOQ6qkAuN+lGl/pTxrR7Cz4o27OLhqJMQMiw=;
b=rRHhhft6QXCXyDtPD45+NgJDaynNybmKW2leBWEIcgfO9JyCKL5G4yO9u9+4c3GeGfl3…<truncated>
```

You can also use the DKIM Test tool at: https://www.appmaildev.com/en/dkim. This will give you a one-time email address to send a message to and will provide a full email authentication report.

To check your DKIM selector is correctly configured in DNS, use the MX toolbox Supertool at: https://mxtoolbox.com/SuperTool.aspx. You will need to know the name of the selector you are testing. The format to use is: <domain>.govt.nz:<selector_name>. It will return the version (DKIM1), key length, and public key if they are correctly configured.

Appendix 1 – DKIM Description

What is DKIM

DKIM is an email authentication method designed to detect forged sender addresses in emails. It allows the receiver to check that an email claimed to have come from a specific domain was indeed authorised by the owner of that domain. It also Cryptographically signs emails to ensure the message is not altered in transit.

Key Benefits:

- Verifies the integrity of the message.
- Helps lessen spoofing and phishing.
- Improves email deliverability and trust.
- Ensures the message has not been altered in transit.

How DKIM works

Signing the Email (Sender Side):

- The sending mail server generates a digital signature using a private key.
- This signature is added to the email header as a DKIM-Signature field.
- The signature is based on selected parts of the email (e.g., body, headers).

Publishing the Public Key (DNS):

- The domain owner publishes a public key in a DNS TXT record under a specific selector.

Verifying the Email (Receiver Side):

- When DMARC checks are completed on the receiving server, it extracts the DKIM signature and selector from the email.
- It queries DNS for the public key.
- It uses the public key to verify the signature and confirm the email hasn't been tampered with.
- It compares the domain used in the signing with the domain used in the From address of the email, to confirm alignment (ref Section 5.3 below).

DKIM Alignment

What is DKIM Alignment?

DKIM alignment is a concept used in DMARC (Domain-based Message Authentication, Reporting & Conformance) to determine whether the domain used in the DKIM signature aligns with the domain in the From header of the email.

Why It Matters:

DMARC requires that either DKIM or SPF pass and align. Without alignment, even a valid DKIM signature won't satisfy DMARC.

How DKIM Alignment Works:

When DMARC checks an inbound email for DKIM alignment it is comparing the result of a DNS lookup, on the d= field, to the RFC5322 “From” address within the email. This is the visible From address which the recipient can generally see in their email client, not the envelope sender which is used in SPF.

The DKIM signature contains the From Header Domain tag which appears as d=<domain> for example d=minties.govt.nz.

This is compared to the From Header Domain in the email, which is the domain visible in the sender’s address, e.g. sam.ashveil@**minties.govt.nz**

DMARC checks if the d= domain in the DKIM signature matches (or is a subdomain of) the domain in the **From** header. If Strict Alignment is used (adkim=s in the DMARC record) then the d= must exactly match the From domain. If Relaxed Alignment is used (adkim=r in the DMARC record) then d= can be a subdomain of the From domain.

Here’s some examples:

DKIM d= Domain	From Header Domain	Alignment
minties.govt.nz	minties.govt.nz	✅ Will pass for either Strict or Relaxed alignment
mail.minties.govt.nz	minties.govt.nz	❌ Strict, ✅ Relaxed Strict alignment will fail, Relaxed alignment will pass
anotherparty.com	minties.govt.nz	❌ No Alignment. All alignment checks will fail, DMARC checks for DKIM will not pass this email.

Bulk Senders (e.g., CRMs, ESPs)

When using third-party platforms to send bulk emails, DKIM alignment can be tricky. This is because many platforms sign emails using their own domain by default (e.g. For the SendGrid service by default they appear from d=sendgrid.net).

This fails DKIM alignment unless you configure custom DKIM within the SendGrid platform. Section 4.3 contains an example on how to set this up.

You must configure the platform to sign emails using your domain (e.g., d=minties.govt.nz).

This usually involves:

- Adding DNS records (public key).
- Verifying domain ownership.
- Enabling custom DKIM in the platform.

Multiple Platforms, Multiple Selectors

When using multiple different platforms for sending emails, each platform should be configured with its own DKIM selector and associated DNS record(s). Do not use the same private key for DKIM signing across multiple services.

Reply Handling

Replies often go to your corporate mail system, which uses its own DKIM setup. This is fine as long as your mail system also signs with aligned DKIM.

Best Practices for DKIM Alignment with Bulk Senders

1. Use Custom DKIM Signing:
    - Ensure all platforms sign with your domain (d=minties.govt.nz).
    - Avoid default DKIM from third-party domains.
2. Configure Unique Selectors:
    - Use distinct selectors per platform (e.g., crm2025, support, marketing).
3. Enable Relaxed Alignment in DMARC (but only if needed):
    - This allows subdomains to pass alignment.
4. Monitor with DMARC Reports:
    - Use a DMARC Reporting tool to track alignment issues.
5. Test Before Going Live:
    - Always test for DKIM alignment issues before going live.

Appendix 2 – Frequently asked Questions

Should every email sending service have its own subdomain?

Using separate subdomains for each external mail sending service is best practice.

The benefits of doing so are:

1. It gives you a clear separation of mail streams making it easier to manage authentication, reputation, and troubleshooting through isolating:
    - Transactional emails (e.g., billing.yourdomain.com)
    - Marketing emails (e.g., promo.yourdomain.com)
    - Internal communications (e.g., yourdomain.com)
2. Improved Deliverability and Reputation Management:
    - If a subdomain gets flagged or blacklisted, your primary domain remains unaffected.
    - You can monitor and manage sender reputation separately for each subdomain.
3. Easier DKIM, SPF, and DMARC Configuration:
    - You can delegate DNS control or authentication records to third-party services without exposing your root domain.
    - You can tailor DMARC policies per subdomain (e.g., strict for transactional, relaxed for marketing).
4. Branding and Transparency:
    - Subdomains can reflect the purpose of the email (e.g., alerts.yourdomain.com), which builds trust with recipients.
    - Helps recipients distinguish between different types of communication.

What key length is recommended?

It is recommended to use 2048-bit keys on all systems which support them. You can revert to 1024-bit keys on legacy systems if needed.

It is possible to sign using 1024, 2048 and 4096 bit RSA keys. 1024 is the most widely supported and most commonly used. 2048 provides a greater level of security, however some legacy systems may not support it. 4096 bit keys provide the greatest level of security, however are rarely used due to poor compatibility.

If you have a specific need to use 4096-bit keys ensure you've verified full end-to-end compatibility.

What happens if there is no adkim= record in my DMARC?

If you have configured DKIM signing and do not have an adkim= statement in your DMARC record in DNS it will default to adkim=r for relaxed. The valid entries for adkim= are adkim=s for strict alignment and adkim=r for relaxed alignment

Can DKIM pass but DMARC still fail?

Yes. DKIM can pass cryptographic validation, but DMARC will fail if:

- The d= domain in DKIM does not align with the "From" domain (based on adkim policy).

- SPF fails and DKIM fails alignment.
- The message is missing a valid DKIM signature altogether.

What if my DKIM signature uses a third-party domain (e.g., d=mcsv.net)?

This will not align with your "From" domain unless you use CNAME delegation, allowing the DKIM signature to appear as if it's from your domain.

Can one email be DKIM signed multiple times?

Yes, this is perfectly valid and a common scenario. An email coming from M365 where the sending Agency uses an external mail security platform will likely be signed at both locations.

The recipient server only needs 1 DKIM record to pass DKIM checks to verify the email.

Appendix 3 – List of Acronyms

Acronym	Definition
CNAME	Canonical Name (used as an alias in DNS)
CRM	Customer Relationship Management
DKIM	Domain Keys Identified Mail
DMARC	Domain-based Message Authentication, Reporting, and Conformance
DNS	Domain Name System
ESP	Email Service Providers (Typically referring to bulk senders)
MX	Mail Exchange
M365	Microsoft 365
NZISM	New Zealand Information Services Manual
SGE	Secure Government Email (Framework)
SPF	Sender Policy Framework