

SAP-ERP Portal — DNS & HTTPS Setup Guide

Internal reference: friendly hostnames (erp.mipl.com, app1.mipl.com, app2.mipl.com) via internal DNS + Apache reverse proxy, secured with a certificate issued by the internal MIPL Active Directory Certificate Authority.

Environment

Component	Machine / Address	Notes
Application server	10.9.11.210	Runs the Node.js portal (PM2, port 5000) and XAMPP Apache (reverse proxy)
Internal DNS server	192.9.201.201 (adc.mipl.com)	Windows Server — AD DS + DNS + AD CS roles
Internal Certificate Authority	adc.mipl.com	Issues the HTTPS certificate used by Apache
Internal domain / zone	mipl.com	Forward lookup zone on the DNS server
Target hostnames	erp.mipl.com, app1.mipl.com, app2.mipl.com	All resolve to 10.9.11.210 via a wildcard DNS record

How it fits together

A browser resolves **erp.mipl.com** to 10.9.11.210 via internal DNS, then connects to Apache on port 80/443. Apache terminates the connection (and, for HTTPS, the TLS handshake) and silently forwards the request to the Node.js portal listening on **127.0.0.1:5000**, managed by PM2. This is what removes the need to type a port number, and lets several apps share the same server on the same ports, routed purely by hostname.

Part A — DNS Configuration

A1. Add a wildcard DNS record

On the DNS server (192.9.201.201 / adc.mipl.com):

- 1 Open **Server Manager** → **DNS**, right-click the server → **DNS Manager** (or run `dnsmgmt.msc` directly).
- 2 Expand the server → **Forward Lookup Zones** → open the **mipl.com** zone.
- 3 Right-click inside the zone → **New Host (A or AAAA)...**
- 4 Name: * IP address: 10.9.11.210
- 5 Leave "Create associated pointer (PTR) record" unchecked. Click **Add Host** → **Done**.

The wildcard means **any** subdomain of mipl.com (erp, app1, app2, and anything added later) resolves to 10.9.11.210 automatically — no further DNS changes needed when a new app is added, only a new entry in the Apache config (Part B).

A2. Verify DNS resolution

From any domain-joined PC:

```
nslookup erp.mipl.com
# Expected: Address: 10.9.11.210
```

A3. Workgroup (non-domain) PCs

A PC that is not joined to the mipl.com domain will not automatically use 192.9.201.201 as its DNS server. For each such machine, either:

- Set its Preferred DNS server to 192.9.201.201 in the network adapter's IPv4 settings, *or*
- Add explicit entries to `C:\Windows\System32\drivers\etc\hosts`:

```
10.9.11.210    erp.mipl.com
10.9.11.210    app1.mipl.com
10.9.11.210    app2.mipl.com
```

Part B — Apache Reverse Proxy (removes the port number)

This is done on **10.9.11.210**, using the Apache that ships with XAMPP (D:\apps\xampp74\apache). It listens on port 80/443 and silently forwards to the Node app on port 5000 (managed separately by PM2 — Apache does not need to know or care how the Node process is supervised).

B1. Enable the proxy modules

Edit D:\apps\xampp74\apache\conf\httpd.conf and remove the leading # from **both** of these lines (both are required — missing just mod_proxy_http causes an "AH01144: No protocol handler" error later):

```
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_http_module modules/mod_proxy_http.so
```

Also confirm this line is present and uncommented (usually further down the same file):

```
Include conf/extra/httpd-vhosts.conf
```

B2. Add a virtual host per application

Edit D:\apps\xampp74\apache\conf\extra\httpd-vhosts.conf:

```
<VirtualHost *:80>
    ServerName erp.mipl.com
    ProxyPreserveHost On
    ProxyPass / http://127.0.0.1:5000/
    ProxyPassReverse / http://127.0.0.1:5000/
</VirtualHost>

<VirtualHost *:80>
    ServerName app1.mipl.com
    ProxyPreserveHost On
    ProxyPass / http://127.0.0.1:<APP1_PORT>/
    ProxyPassReverse / http://127.0.0.1:<APP1_PORT>/
</VirtualHost>

<VirtualHost *:80>
    ServerName app2.mipl.com
    ProxyPreserveHost On
    ProxyPass / http://127.0.0.1:<APP2_PORT>/
    ProxyPassReverse / http://127.0.0.1:<APP2_PORT>/
</VirtualHost>
```

Replace <APP1_PORT> / <APP2_PORT> with the actual ports those applications listen on.

B3. Port 80 must be free, and open on the firewall

- 1 Confirm nothing else on 10.9.11.210 is already using port 80 (IIS, another web server, etc.) — netstat -ano | findstr :80
- 2 Open the port on Windows Firewall: netsh advfirewall firewall add rule name="HTTP 80" dir=in action=allow protocol=TCP localport=80
- 3 Restart Apache from the XAMPP Control Panel (Stop, then Start — not just Reload).

B4. Test

```
http://erp.mipl.com    (no port number)
```

Should load the portal exactly as <http://10.9.11.210:5000> did before.

Part C — HTTPS Certificate (Internal CA)

Because **mipl.com** is an internal-only name, a public certificate authority (like Let's Encrypt) cannot issue a certificate for it. Instead, the certificate is issued by the organisation's own internal CA (Active Directory Certificate Services, running on `adc.mipl.com`). Domain-joined PCs trust this CA automatically; workgroup PCs need one manual step (Part D).

C1. Allow the certificate to carry Subject Alternative Names (SAN)

By default, a Windows CA ignores any Subject Alternative Name supplied in a certificate request, for security reasons. Modern browsers **require** a matching SAN and ignore the plain Common Name — skipping this step causes a `NET::ERR_CERT_COMMON_NAME_INVALID` error later. On the CA server (`adc.mipl.com`), as Administrator:

```
certutil -setreg policy\EditFlags +EDITF_ATTRIBUTESUBJECTALTNAME2
net stop certsvc
net start certsvc
```

C2. Grant Enroll permission on the Web Server template

The built-in **Web Server** certificate template only allows Domain/Enterprise Admins to enroll by default. On the CA server, run `certtmpl.msc` → right-click **Web Server** → **Properties** → **Security** → add the relevant user or computer account → allow **Enroll** (and Read).

C3. Generate a WILDCARD private key + CSR on 10.9.11.210

Use a wildcard certificate (`*.mipl.com`) rather than one listing each hostname individually. With a wildcard DNS record (Part A1) and a wildcard certificate, adding `app3`, `app4`, ... `appN` later needs **no new DNS record and no new certificate at all** — see Part E. XAMPP's bundled OpenSSL looks for its config in a hardcoded default path that doesn't exist on this install — point it at the real one first:

```
cd D:\apps\xampp74\apache\bin
set OPENSSL_CONF=D:\apps\xampp74\apache\conf\openssl.cnf

openssl req -new -newkey rsa:2048 -nodes -keyout wildcard.mipl.com.key -out wildcard.mipl.com.csr ^
  -subj "/CN=*.mipl.com/O=MIPL/C=IN" ^
  -addext "subjectAltName=DNS:*.mipl.com,DNS:mipl.com"
```

This produces `wildcard.mipl.com.key` (private key — stays on this server) and `wildcard.mipl.com.csr` (the request). This one certificate covers `erp.mipl.com`, `app1.mipl.com` ... `appN.mipl.com` — anything under `mipl.com`.

Internal CAs (Microsoft ADCS) don't enforce the wildcard restrictions public CAs do, so this works the same way a single-name request did. If your organisation's policy prefers explicit names instead of a wildcard, list them all with `-addext "subjectAltName=DNS:erp.mipl.com,DNS:app1.mipl.com,..."` — but then every new app needs the CSR/certificate steps repeated.

C4. Submit the request to the internal CA

```
certreq -submit -attrib "CertificateTemplate:WebServer" wildcard.mipl.com.csr wildcard.mipl.com.crt
```

A dialog appears to pick the CA (auto-discovered via AD) — select the one on `adc.mipl.com`. If issued immediately, the signed certificate is saved as `wildcard.mipl.com.crt`. If it instead returns "Pending", an administrator must approve it on the CA (Certification Authority console → Pending Requests → right-click → Issue), then retrieve it:

```
certreq -retrieve <RequestID> wildcard.mipl.com.crt
```

C5. Export the CA's root certificate

On the CA server: `certsrv.msc` → right-click the CA name → **Properties** → **General** → **View Certificate** → **Details** → **Copy to File** → export as **Base-64 encoded X.509 (.CER)**. Save it as `ca-root.crt` and copy it to 10.9.11.210.

C6. Install the certificate in Apache

- 1 Create the folder and move the files in: D:\apps\xampp74\apache\conf\ssl\ should contain wildcard.mipl.com.crt, wildcard.mipl.com.key and ca-root.crt (the CA export from C5 — whatever extension it was saved with, e.g. .cer, is fine as long as the filename in the config below matches it exactly).
- 2 In httpd.conf, uncomment:

```
LoadModule ssl_module modules/mod_ssl.so
Include conf/extra/httpd-ssl.conf
```

Add one SSL virtual host **per app** in httpd-vhosts.conf — every block reuses the **same** wildcard cert/key files, only ServerName and the backend port change:

```
<VirtualHost *:443>
    ServerName erp.mipl.com
    SSLEngine on
    SSLCertificateFile "D:/apps/xampp74/apache/conf/ssl/wildcard.mipl.com.crt"
    SSLCertificateKeyFile "D:/apps/xampp74/apache/conf/ssl/wildcard.mipl.com.key"
    SSLCertificateChainFile "D:/apps/xampp74/apache/conf/ssl/ca-root.crt"

    ProxyPreserveHost On
    ProxyPass / http://127.0.0.1:5000/
    ProxyPassReverse / http://127.0.0.1:5000/
</VirtualHost>

<VirtualHost *:443>
    ServerName app1.mipl.com
    SSLEngine on
    SSLCertificateFile "D:/apps/xampp74/apache/conf/ssl/wildcard.mipl.com.crt"
    SSLCertificateKeyFile "D:/apps/xampp74/apache/conf/ssl/wildcard.mipl.com.key"
    SSLCertificateChainFile "D:/apps/xampp74/apache/conf/ssl/ca-root.crt"

    ProxyPreserveHost On
    ProxyPass / http://127.0.0.1:3001/
    ProxyPassReverse / http://127.0.0.1:3001/
</VirtualHost>
```

Make sure only **one** "Listen 443" line exists across the whole config (it already exists by default in httpd-ssl.conf — adding a second one anywhere else stops Apache from starting).

C7. Firewall + restart + test

- 1 Open port 443: netsh advfirewall firewall add rule name="HTTPS 443" dir=in action=allow protocol=TCP localport=443
- 2 Restart Apache from the XAMPP Control Panel.
- 3 If it fails to start, run httpd.exe -t from D:\apps\xampp74\apache\bin to print the exact config error.
- 4 Test from a domain-joined PC: https://erp.mipl.com — should show a trusted padlock, no warning.

Part D — Workgroup (Non-Domain) PCs

Domain-joined PCs trust the internal CA automatically. A PC that is **not** joined to the mipl.com domain never receives that trust automatically, and will always show a security warning until it is set up manually — once per machine.

D1. Name resolution (see also Part A3)

Either point its DNS server at 192.9.201.201, or add hosts-file entries for each app hostname.

D2. Import the CA root certificate (required for HTTPS to show as trusted)

Copy `ca-root.crt` to the machine, then as Administrator:

```
certutil -addstore -f "ROOT" ca-root.crt
```

D3. Firefox needs one extra step

Firefox keeps its own certificate store — it does not read the Windows trust store by default, so step D2 alone will not satisfy Firefox. Either:

- **Recommended:** in Firefox, open `about:config`, set `security.enterprise_roots.enabled` to `true`, restart Firefox. (This can be pushed to many PCs at once via a Firefox enterprise policy file / GPO ADMX template.)
- **Or** import `ca-root.crt` directly into Firefox: Settings → Privacy & Security → Certificates → View Certificates → Authorities → Import → check "Trust this CA to identify websites".

Part E — Adding a New App (app2, app3, ... appN)

Because Parts A-C already set up a **wildcard** DNS record and a **wildcard** certificate, adding another app under `mipl.com` needs none of that repeated. It's a two-line addition to Apache's config, done entirely on 10.9.11.210.

E1. Confirm the app has a backend port

The new app must already be running and listening on its own port (however it's managed — PM2, a Windows service, IIS, etc.). Confirm with, e.g. for port 3002:

```
netstat -ano | findstr :3002
```

You should see a LISTENING line. If not, start that app first — Apache can't proxy to a port nothing is listening on.

E2. Add two virtual host blocks

In `D:\apps\xampp74\apache\conf\extra\httpd-vhosts.conf`, add an HTTP block (for plain `http://` access / redirects) and an HTTPS block reusing the existing wildcard cert:

```
<VirtualHost *:80>
    ServerName app2.mipl.com
    ProxyPreserveHost On
    ProxyPass / http://127.0.0.1:3002/
    ProxyPassReverse / http://127.0.0.1:3002/
</VirtualHost>

<VirtualHost *:443>
    ServerName app2.mipl.com
    SSLEngine on
    SSLCertificateFile "D:/apps/xampp74/apache/conf/ssl/wildcard.mipl.com.crt"
    SSLCertificateKeyFile "D:/apps/xampp74/apache/conf/ssl/wildcard.mipl.com.key"
    SSLCertificateChainFile "D:/apps/xampp74/apache/conf/ssl/ca-root.crt"

    ProxyPreserveHost On
    ProxyPass / http://127.0.0.1:3002/
    ProxyPassReverse / http://127.0.0.1:3002/
</VirtualHost>
```

E3. Restart and test

- 1 Check syntax first: `httpd.exe -t` from `D:\apps\xampp74\apache\bin` — should print "Syntax OK".
- 2 Restart Apache from the XAMPP Control Panel.
- 3 Confirm DNS resolves first: `nslookup app2.mipl.com` should return 10.9.11.210 automatically (the wildcard record from Part A1 covers it — no new DNS entry needed).
- 4 Test `http://app2.mipl.com` and `https://app2.mipl.com`.

Nothing outside Apache's own config file needs to change for a new app — no new DNS record, no new certificate request, no new firewall rule (80/443 are already open). That's the entire benefit of doing the wildcard setup once in Parts A and C.

Troubleshooting Log

Issues actually hit while setting this up, and their fixes — kept here for next time.

Symptom	Cause	Fix
Apache: "Invalid command 'ProxyPreserveHost'"	mod_proxy / mod_proxy_http not loaded	Uncomment both LoadModule lines in httpd.conf (Part B1)
Browser: 500 Internal Server Error; log shows AH01144 "No protocol handler"	mod_proxy loaded but mod_proxy_http still commented out	Uncomment LoadModule proxy_http_module specifically
openssl: "Can't open C:\Apache24\conf\openssl.cnf"	OpenSSL's compiled-in default config path doesn't match this XAMPP install	set OPENSSL_CONF=D:\apps\xampp74\apache\conf\openssl.cnf before running openssl
certreq: "Denied by Policy Module... permissions on the certificate template"	Requesting identity has no Enroll permission on the Web Server template	Grant Enroll on the template (Part C2); if the machine is not domain-joined, submit the CSR from a domain-joined machine instead
Apache fails to start after adding the SSL vhost, no new error.log entries	Config file references a certificate file that doesn't exist at that exact path (e.g. wrong extension)	Run "httpd.exe -t" from apache\bin to print the exact syntax/path error
Browser: NET::ERR_CERT_AUTHORITY_INVALID	This PC/browser doesn't trust the issuing CA yet	Domain PC: gpupdate /force, or check certmgr.msc has the CA under Trusted Root CAs. Workgroup PC: certutil -addstore -f ROOT ca-root.crt (Part D2)
Browser: NET::ERR_CERT_COMMON_NAME_INVALID	Certificate has no Subject Alternative Name matching the hostname (CN alone is not accepted by modern browsers)	Enable SAN on the CA (Part C1) and reissue the certificate with -addext "subjectAltName=..." (Part C3)
Works in Chrome/Edge but not Firefox, same PC	Firefox uses its own certificate store, not the Windows one	Enable security.enterprise_roots.enabled in about:config, or import the CA cert into Firefox directly (Part D3)
New app's hostname (e.g. app1.mipl.com) doesn't load, but erp.mipl.com works fine on the same server, Apache syntax check passes, and the backend port is confirmed listening	DNS never actually has a record for the new hostname — often only an explicit 'erp' host record was added, not the wildcard '*' record from Part A1	Run "nslookup app1.mipl.com" and compare to "nslookup erp.mipl.com". If the new name returns Non-existent domain, add the wildcard A record (* -> 10.9.11.210) in DNS Manager so every subdomain resolves automatically
Apache Control Panel shows a vague "shutdown unexpectedly" message with no useful detail	The XAMPP Control Panel log is not the real error source	Always run "httpd.exe -t" from apache\bin directly — it prints the exact syntax error and line number immediately, instead of a generic Control Panel message

Prepared as an internal setup reference for the SAP-ERP portal (mipl.com internal network).