Introduction

AirPrint

AirPrint is Apple's marketing name for the "IPP Everywhere" extensions to the Internet Printing Protocol (IPP). AirPrint is a driverless printing protocol that takes advantage of the additional computing capability of modern printers to move the "driver" from the client device into the printer itself.

Bonjour

Bonjour is Apple's marketing name for the combined mDNS (Multicast DNS) and DNS-SD (DNS Service Discovery) IETF standard protocols. These protocols work together to allow devices to discover and connect to each other on small and large networks with minimal or no configuration. AirPrint relies on Bonjour in order to discover nearby printers.

Most frequently, "Bonjour" refers to the mDNS (or "multicast") mode of Bonjour, while DNS-SD (the "unicast" or "managed" mode) is deployed much less frequently. DNS-SD is required when client devices and servers are on different network segments, and is deployed on large school or enterprise networks.

Both modes of Bonjour use the standard DNS (Domain Name System) packet format. Each advertised "service" consists of a PTR and SRV record to describe the service type, name, host, and port number, along with a TXT record describing additional service-specific attributes. Service "discovery" involves querying for these records under one or more domains. The "local" domain indicates multicast DNS shall be used to resolve names, while any other domain results in standard DNS queries.
**mDNS (Multicast DNS)**

The local network operating mode of Bonjour is defined by IETF RFC 6762, "Multicast DNS", and is designed for use on a single, layer 2 or "link local" network segment. This mode of operation can be thought of as "peer to peer" as it requires no centralized DNS infrastructure and is truly zero configuration.

This mode of Bonjour uses the same packet format as standard DNS, but instead of sending queries directly to the configured name server(s), mDNS uses multicast to send queries to all nodes on the network. This allows Bonjour to work on simple home and small business networks with no configuration required. Devices that provide the service being queried respond, while others simply ignore it.

Multicast Bonjour uses the ".local" top level domain to place local services in their own unique namespace.

---

**Technical Information**

<table>
<thead>
<tr>
<th>Protocol</th>
<th>UDP port 5353</th>
</tr>
</thead>
</table>
| Multicast Addresses| 224.0.0.251 (01:00:5E:00:00:FB)  
|                   | ff02::fb (33:33:00:00:00:FB)     |
| Packet Size       | Often larger than 512 bytes, up to the link MTU |
| TTL               | 0 to prevent packets from being routed |

---

**When Should I Use It?**

Multicast Bonjour (mDNS) was designed to operate on small networks with a single "link local" network segment. mDNS requires no configuration on the part of the client, server, or network infrastructure. mDNS uses multicast packets with a TTL of zero that are not intended to propagate beyond the local network segment.

Wide Area Bonjour (DNS-SD) was designed to provide the same simple, automatic discovery of network services provided by mDNS across large scale networks. This mode of Bonjour uses standard DNS lookups to discover services anywhere on your network, across multiple segments, and can be thought of as the "managed" version of Bonjour.

*If your network is segmented, we strongly recommend using our Wide Area Printing implementation (based on DNS-SD) instead of Multicast DNS (mDNS). Unfortunately we are not able to support customers attempting to deploy Multicast DNS across more than one network segment as this requires in-depth knowledge of the capabilities and configuration of your specific make and model of network equipment.*

Multicast DNS (mDNS) is not intended to be used across subnets, and may not work properly with enterprise-grade network switches for several reasons (see the troubleshooting section). Additionally, many equipment vendors offer products that attempt to "bridge" Bonjour traffic...
between subnets, but because there is no part of the spec that defines or permits such behavior, every manufacturer implements this differently and with varying degrees of success. We have tested several of these solutions both in our lab and at customer sites and found that while some work okay with Apple TVs, printers and other devices that share a small handful of services, none that we’ve seen implement enough of the specification to work properly with services like Printopia. (Note: These solutions do have their place— they exist so that administrators don’t have to manually enter DNS records for Apple TVs and other devices on their networks.)

Implementing multicast on a large network requires that every single switch, router, controller, and wireless access point between the server and the client device be configured to pass multicast packets correctly. Any change to network settings or a firmware upgrade on any device along the way can affect the operation of multicast Bonjour on your network.

**In Summary**

1. If your devices and servers are all on the same Layer 2 network segment, multicast DNS should work in most situations. Some enterprise grade switches may need to be reconfigured for proper operation.

2. If your network is segmented, use Wide Area Printing (DNS-SD). We strongly recommend avoiding multicast DNS on segmented networks, regardless of any Bonjour proxy, gateway, or reflector solutions you may have.

3. Wide Area Bonjour (DNS-SD) works independently of multicast DNS, and peacefully coexists with any Bonjour proxy, gateway, or reflectors on the network.

**Theory of Operation**

To discover printers on the local network segment a device on the network will issue a lookup request for a PTR record with the name "_ipp._tcp.local". The "_ipp._tcp" indicates we’re looking for IPP (Internet Printing Protocol) services, and the ".local" top level domain tells the resolver to use multicast (mDNS) to resolve the services on the local network segment.

Devices listening on the network that provide IPP printing services will respond to this query with additional information including the name of the service, the host and port number, and a TXT record containing service attributes.

For example, the service "TestPrinter" shared from the host "Mac" would respond with the following set of records:

<table>
<thead>
<tr>
<th>Name</th>
<th>RTYPE</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>_ipp._tcp.local.</td>
<td>PTR</td>
<td>TestPrinter@Mac._ipp._tcp.local.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><em>Says there's a &quot;TestPrinter&quot; service of type &quot;_ipp._tcp&quot; on the host &quot;Mac.local&quot;.</em></td>
</tr>
</tbody>
</table>
To illustrate, you may try this yourself on your network using the "dig" command:

```
dig -p 5353 @224.0.0.251 _ipp._tcp.local ptr
```

The dig command is not aware of the mDNS protocol so it will only return a single response at random out of all responders.
Troubleshooting

Recommended Tools

tcpdump

To view multicast DNS traffic:

```
sudo tcpdump -vvv -s 0 -l -n port 53
```

dig (Domain Information Groper) – Mac, Linux, Windows

The "dig" command is invaluable for diagnosing DNS issues. It ships with Macs, and can be easily installed on Linux or Windows. (To install on Windows, see the ISC BIND website). This command bypasses the operating system's resolver and allows you to direct queries to specific DNS servers on your network.

```
dig [@server] [-p port#][-4][-6] [name] [type]
```

Bonjour Browser – iOS, Mac

Install the free "Discovery - Bonjour Browser" app from Tildesoft, available on the iOS App Store:

An older, slightly less capable version of this app is available for the Mac as well. Google "Tildesoft" to find it.

DNS-SD (Multicast DNS & DNS Service Discovery Test Tool) – Mac

The "dns-sd" tool allows you to browse for services and list the currently discovered "Bonjour Browse Domains" using the command line.

```
dns-sd -E          (Enumerate recommended registration domains)
dns-sd -F          (Enumerate recommended browsing domains)
dns-sd -R <Name> <Type> <Domain> <Port> [TXT]... (Register a service)
dns-sd -B <Type> <Domain> (Browse for services instances)
dns-sd -L <Name> <Type> <Domain> (Look up a service instance)
dns-sd -P <Name> <Type> <Domain> <Port> <Host> <IP> [TXT]... (Proxy)
dns-sd -q <name> <rtype> <rrclass> (Generic query for any record type)
dns-sd -D <name> <rtype> <rrclass> (Validate query for any record type with DNSSEC)
dns-sd -Z <Type> <Domain> (Output results in Zone File format)
dns-sd -G v4/v6/v4v6 <name> (Get address information for hostname)
dns-sd -g v4/v6/v4v6 <name> (Validate address info for hostname with DNSSEC)
dns-sd -V              (Get version of currently running daemon / system service)
```
Wireshark is a packet analyzer that will show you exactly what's being sent over your network. Bonjour packets make use of the standard DNS packet format and are well supported by WireShark.
Common Symptoms

Printer sharing seems to work for a while, after which printers disappear from the network. Turning sharing off and on again seems to fix the problem for a while, only to stop working again later.

This happens when multicast is not working properly on the network, resulting in unidirectional communication where clients can receive broadcasts from the server, but the server cannot receive broadcasts from clients. When a service is first shared, the server broadcasts the new service to all clients able to listen. These records are then cached client-side.

After these records expire, the services will disappear from client devices. When the client wants to discover these services again, it will broadcast a request for them on the network, but for some reason these requests are blocked and do not make it back to the server.

The solution can be complex. Getting multicast working reliably on a large network means configuring every network switch along the path between the two devices to properly handle multicast traffic.

First of all, the address 224.0.0.251 should be exempt from IGMP. This address should always be broadcast to every port on a switch regardless of IGMP group membership, the same way other intra-switch protocols including OSPF are handled. However, we've seen this handled differently by almost every vendor out there. You may need to manually configure each switch to pass Bonjour to all ports.

Many network vendors apply additional controls over multicast traffic.

Services are slow to resolve.

This can happen on networks that pass IPv4 multicast traffic okay, but not IPv6 multicast traffic, in cases where the server is assigned an IPv6 address. Network clients may try to resolve the service using IPv6, only to timeout and try again over IPv4. The IPv6 equivalent to IGMP is MLD, which also needs to be configured on every switch.

If you're unsure if your network is passing IPv6 properly, disable IPv6 entirely on the server running Printopia Pro.

The printer list is incomplete at times.

Multicast over Wi-Fi can be unreliable under poor signal conditions, or at the far end of the range envelope of a wireless access point. You may need to adjust the multicast rate of your wireless radio to optimize performance in your environment.

You may also switch to DNS-SD which uses unicast, avoiding these issues.
Common Causes

Intentional Blocking.

Some networks intentionally prevent multicast Bonjour from working. Make sure you're able to discover other services on your network by using the "Discovery - Bonjour Browser" app on your device. If you see nothing at all, something may be preventing Bonjour from working altogether. Also, some networks block peer-to-peer connectivity over Wi-Fi. In this case, wireless clients can see the internet and other wired clients/servers, but not other wireless clients. We do not recommend running the Printopia Pro server over wireless.

Firewalls.

Many older firewalls incorrectly classify DNS packets larger than 512 bytes in length as malicious. This can break DNSSEC and Bonjour, and any other service that depends on large DNS packets.

This bad advice was given even recently in response to CVE-2015-7547, the glibc getaddrinfo stack-based buffer overflow vulnerability disclosed in February 2016.

Make sure any firewall is configured to pass mDNS (UDP port 5353) traffic to avoid potential problems. The firewall should not discard UDP fragments, and should not limit the maximum packet size.

Make sure the addresses 224.0.0.251 and ff02::fb (33:33:00:00:00:FB) are permitted as well.

Microsoft used to recommend use of ".local" as a pseudo-TLD for use on private networks with internal DNS servers.

This conflicts with the Bonjour definition of ".local", which interprets the domain to mean "resolve this domain using multicast". Microsoft has long since stopped recommending the use of this TLD. On networks where this TLD is still in use, Bonjour has a coexistence mechanism in place but we have not tested it.

Bonjour Gateways.

Bonjour "gateways", "proxies" and "reflectors" are products intended to make Bonjour work across subnets. Since the specification forbids such things, there is no standard defined to make this possible, so every vendor implements theirs differently, with varying degrees of success. We have tested several of these solutions both in our lab and at customer sites and found that while some work okay with Apple TVs, printers and other devices that share a small handful of services, none that we've seen implement enough of the specification to work properly with services like Printopia.

These hacks avoid the need to centrally administer Apple TVs and similar devices on the network by manually adding records to a central name server, but they simply don't work reliably in our experience with devices that share a large number of services.
(In our testing it appears as though many of these solutions don't handle MTU issues properly, and in other cases improperly cache truncated partial responses as complete responses.)

Printopia Pro's Wide Area Printing feature works completely independently of multicast (and any Bonjour proxy on your network), so the two can coexist peacefully.

**Misconfigured Routers/Switches.**

Getting multicast working reliably on a large network means configuring every network switch along the path between the two devices to properly handle multicast traffic. Every network vendor handles multicast differently, adding their own optimizations and quirks.

Bonjour traffic should be exempt from IGMP, but many switches do not honor this. Check that IGMP and MLD are configured properly, or see if you can bypass them and always forward Bonjour's addresses to all switch ports.
Troubleshooting Summary

1. **If the Mac running Printopia Pro is not on the same Layer-2 network segment as your wireless clients, we strongly urge you to use our "wide area printing" feature.** Getting multicast to work reliably on a large network, and to survive network equipment upgrades and firmware updates is challenging at best. We do not recommend the use of Bonjour proxies, gateways, reflectors, or other related products. Bonjour provides a standard, robust way to operate across subnets using the DNS-SD protocol that avoids all of the pitfalls relating to multicast. **Unfortunately, we cannot support you if you are attempting to use a Bonjour gateway, proxy, or reflector. Since there is no specification that defines their operation, every manufacturer has implemented theirs differently, with varying degrees of success. Additionally, based on our own in-house evaluation and testing at customer sites, while these solutions can work okay with Apple TVs and other simple devices, they often do not support enough of the protocol to work properly with devices that share a large number of services. Printopia falls into this category, as does any print server.**

2. **If you're running Mac OS X 10.10 through 10.10.3 on your server, upgrade to 10.10.4 or later.** The primary purpose of the 10.10.4 update was to address some catastrophic networking issues affecting Bonjour and multicast DNS.

3. **Make sure the Printopia Pro server is accessible from your wireless network.** Use a network utility on the iOS device to ping the server, and also verify you can connect using Safari to the URL [https://serverip:10631](https://serverip:10631). (Replace serverip with the IP address of the server, and 10631 with the port you've chosen if you changed it for some reason). If this fails, try enabling DHCP on the Printopia Pro server instead of using a static IP address.

4. **If your server has IPv6 enabled, make sure your network properly handles multicast IPv6.** If you're unsure, and don't need IPv6 enabled, disable it on your server using the "networksetup -setv6off Ethernet" command.

5. **If your Printopia Pro server is connected to Wi-Fi, which we do not recommend, make sure your wireless network does not block "peer to peer" connectivity.** Try pinging between two Wi-Fi connected hosts to test this.