

KIP-602: Change default value for `client.dns.lookup`

- [Status](#)
- [Motivation](#)
- [Proposed Changes](#)
- [Compatibility, Deprecation, and Migration Plan](#)
- [Rejected Alternatives](#)

Status

Current state: "Accepted"

Discussion thread: [here](#)

JIRA:

A screenshot of a JIRA error message. It features a yellow warning triangle icon on the left, followed by the text "Unable to render Jira issues macro, execution error." in a blue font.

Please keep the discussion on the mailing list rather than commenting on the wiki (wiki discussions get unwieldy fast).

Motivation

[KIP-302](#) introduced `"use_all_dns_ips"` value for `client.dns.lookup` configuration to make the NIO client trying all the possible IP's of a hostname before failing the connection to that hostname.

It explicitly rejects making `"use_all_dns_ips"` as the default to avoid impacting existing users, but it did not explain what the impact is.

With this KIP, we would like to argue that it is desirable to make `"use_all_dns_ips"` as the default value for `client.dns.lookup` for these reasons:

1. reduce connection failure rates by using all the possible IP addresses of a hostname
2. common expectation of applications dealing with hostname resolved to multiple IP addresses is to attempt connecting to all of them and use the one it can connect successfully, not to use the first one
3. resolving hostname to multiple IP addresses is becoming more common due to the rise of cloud and containerised environment

Public Interfaces

The default value of `client.dns.lookup` changes from `"default"` (using the first resolved IP) to `"use_all_dns_ips"` (using one of the resolved IP's, whichever gets successful connection first).

To achieve the old behaviour of using the first resolved IP, explicitly set the value of `client.dns.lookup` configuration to `"default"`.

Note that this KIP preserves [KIP-302](#) behaviour to only use multiple IPs of the same type (IPv4/IPv6) as the first one, to avoid any change in the network stack while trying multiple IPs.

With this KIP, client connection to bootstrap server will behave as per the following based on the value of `client.dns.lookup` configuration:

- If set to `"use_all_dns_ips"`, connect to each returned IP address in sequence until a successful connection is established. After a disconnection, the next IP is used. Once all IPs have been used once, the client resolves the IP(s) from the hostname again (both the JVM and the OS may cache DNS name lookups, however).
- If set to `"resolve_canonical_bootstrap_servers_only"`, expands each bootstrap address into a list of canonical names. After the bootstrap phase, each of the canonical names is resolved with the same behaviour as `"use_all_dns_ips"`.
- If set to `"default"` (deprecated), attempt to connect to the first IP address returned by the lookup, even if the lookup returns multiple IP addresses.

Proposed Changes

The behaviour of `ClientUtils#resolve` stays the same, which is:

- If `client.dns.lookup` value is `"use_all_dns_ips"` or `"resolve_canonical_bootstrap_servers_only"`: Attempt connecting to each resolved IP addresses and use the first one that connects successfully.
- If `client.dns.lookup` value is `"default"`: Use the first resolved IP address.

Change the default value of `client.dns.lookup` configuration in `AdminClientConfig`, `ProducerConfig` and `ConsumerConfig` from `"default"` to `"use_all_dns_ips"`.

Change all server, tool and test code that literally use `"default"` value to use `"use_all_dns_ips"` instead.

Print a warning if `"default"` value is used: "The 'client.dns.lookup' value 'default' is deprecated and will be removed in future version."

Compatibility, Deprecation, and Migration Plan

- The value `"default"` will be marked as deprecated. And this value will be removed in version 3.0.
- If `client.dns.lookup` is set to `"resolve_canonical_bootstrap_servers_only"`, it will use the behaviour of `"use_all_dns_ips"` when resolving the individual broker address, instead of using the first resolved IP address.

Please note that the concern about breaking SSL hostname verification raised in [KIP-235](#) does not apply to this KIP because this KIP does not propose to modify `ClientUtils#parseAndValidateAddresses`. This KIP is proposing to only modify `ClientUtils#resolve`, which is used in `ClusterConnectionStates#currentAddress`, which is then used in `NetworkClient#initiateConnect` to establish the socket connection to the broker. In other words, this KIP only change the behaviour of resolving address to the IP address used by the socket to connect to the broker.

Rejected Alternatives

- Keep the default behaviour of `client.dns.lookup` to connect to the first resolved IP. The reason this is rejected is because it does not match with the common expectation when a hostname resolves to multiple IP addresses.