BeyondTrust

Remote Support
How to Use Session Policies

TC:2/8/2019

Table of Contents

How to Use Support Session Policies

With session policies, you can customize support session permissions to fit specific support scenarios. A session policy is a set of rules that defines which tools are available in a support session as well as which actions prompt the customer for acceptance.

Use session policies to change the permissions allowed in a support session based on the support portal the customer came through or even the specific endpoint being supported. Session policies provide flexibility in building the security model for each specific support scenario.

Why Use a Session Policy?

For organizations that support a diverse group of clients, representatives need to be able to handle sessions with permissions based on multiple factors. With session policies, a support representative's effective permissions granted and restrictions applied are dynamic, based on criteria other than just that representative's user permissions.

Session policies apply permissions not only to the representative but also to Jump Items and support portals, and by extension to Bomgar Buttons, Jumpoints, and local Jumps. The ability to apply permissions on multiple levels aids in flexibility and security.

Flexibility

With session policies, a representative can be given different permissions for different scenarios. The tools available to a representative and the prompts the customer sees can vary from session to session based not only on the representative account but also on whether the session is customer-initiated or Jump session, on the support portal through which the session started, and on the specific Jump Item. By strategically applying policies, you can set how permissions work in numerous scenarios.

Security

Applying session policies can also help with your security measures. Some of your end-users may be in environments that require stricter security than is required by other end-users. With session policies, you can comply with rigid security requirements without sacrificing use of your full tool set for less-restricted end-users. By assigning separate session policies for different customer support portals, you can adjust the security level based on your clients' needs.

How Do Session Policies Work?

From the /login administrative interface, create session policies to apply to objects within BeyondTrust. Once applied, these policies are layered each time a session starts, and the results of the layered policies determine the permissions for each session. To create session policies, see "How Do I Set Up a Session Policy?" on page 11.

Order for Applying Session Policies

Session policies are applied in a specific order. This order is hard-coded by BeyondTrust and cannot be changed, and higher priority policies cannot be overridden. The order in which policies are applied is:

1. Jump Item
2. Public Portal
3. Representative
4. Global Default

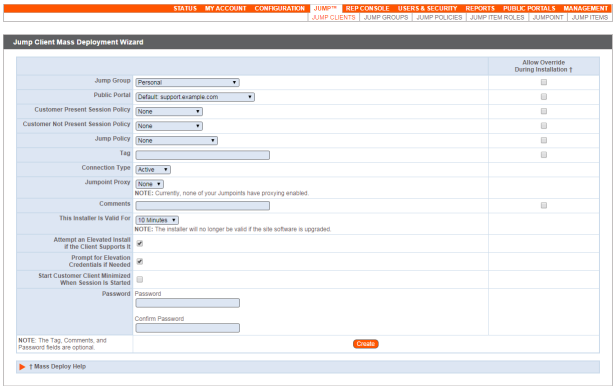
The first applied session policy that defines a setting is used for that setting. If a lower policy also defines that setting, the second policy's setting is ignored.

Any permissions that are left undefined by the first applied policy may be defined by the next policy in the list. If the first applied policy defines all permissions, then none of the lower policies defines any permissions. If no policies are defined for the session, then the global default policy defines all permissions. Because the global default sets all undefined permissions, all permissions within the global default policy must be defined.

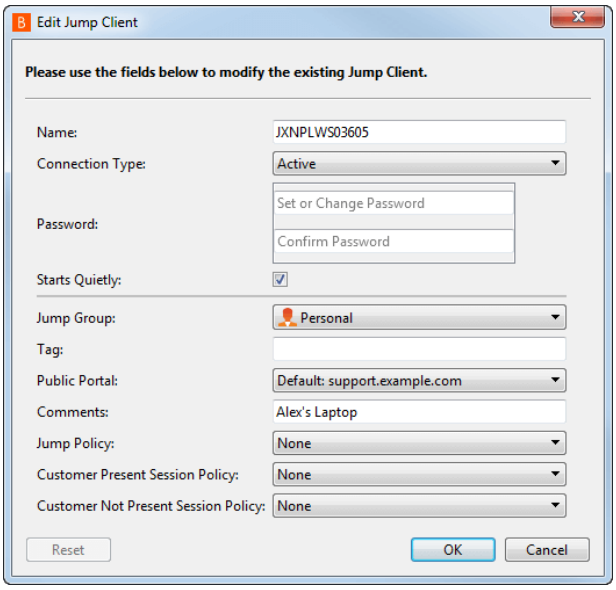
Jump Items

A session policy can be applied to a Jump Client, Local Jump Shortcut, Remote Jump Shortcut, or Shell Jump Shortcut. When a session is started through this Jump Item, the applied session policy sets the permissions that are available within this session. Any permissions that are not defined by the Jump Item session policy can be defined by a lower policy (i.e., public portal policy, representative policy, or global default policy).

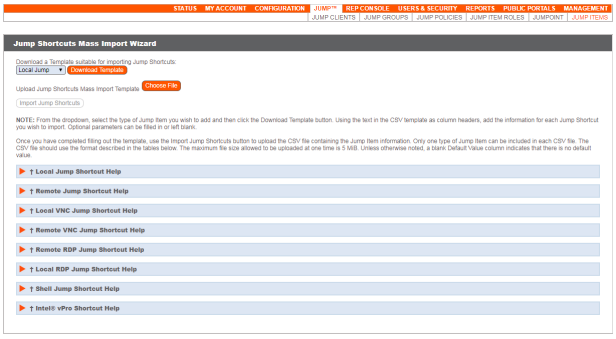
Jump Clients deployed from the mass deployment wizard of the /login interface can be assigned a session policy during creation. You can apply one policy for when the customer is determined to be present and another for when the customer is not present.



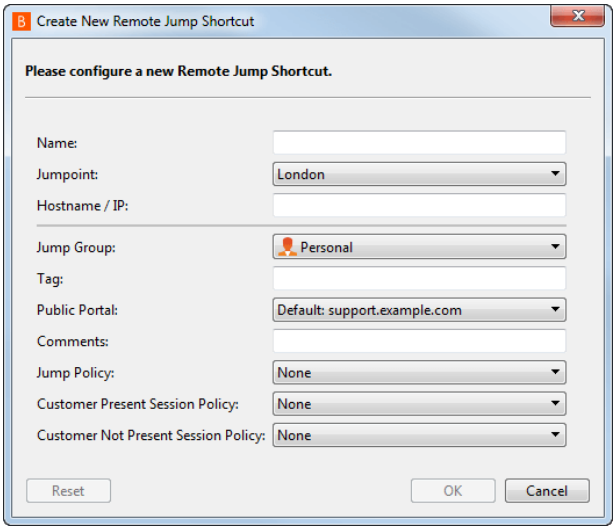
To assign a session policy to a Jump Client deployed within a support session, first customize the Jump Client. Near the bottom of the properties dialog, select a session policy. You can apply one policy for when the customer is determined to be present and another for when the customer is not present.



Local Jump Shortcuts, Remote Jump Shortcuts, and Shell Jump Shortcuts imported via the mass deployment wizard of the /login interface can be assigned a session policy during creation. For Local or Remote Jump Shortcuts, you can apply one policy for when the customer is determined to be present and another for when the customer is not present.



To assign a session policy to a Jump Shortcut deployed from the representative console, scroll to the bottom of the properties dialog. For Remote and Local Jump Shortcuts, you can apply one policy for when the customer is determined to be present and another for when the customer is not present. Shell Jump Shortcuts have only one session policy field.



You can change the session policy for a deployed Jump Item from the Jump interface of the representative console. View the Jump Item properties and select the session policy.

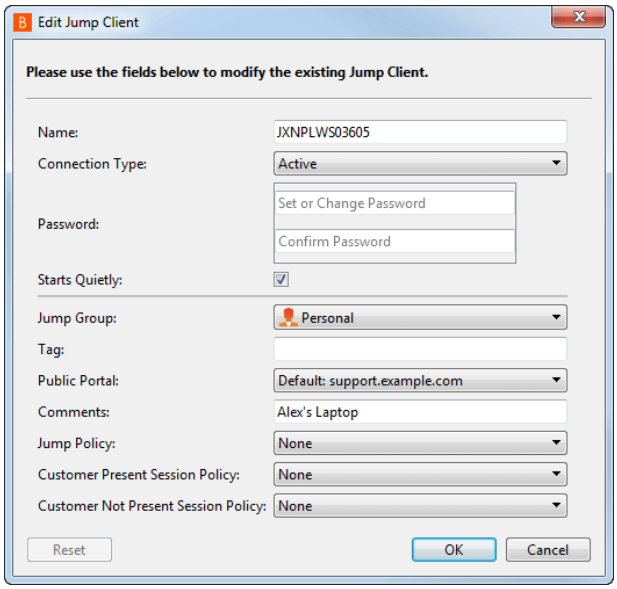
Public Portals

A session policy can also be applied to sessions that come through a specific public portal. This most notably affects customer-initiated sessions. If you have multiple public portals, then the session permissions may differ depending on which public portal your customer used to start a session with you.

Because Jump Items and Bomgar Buttons are associated with public portals, this also affects sessions started through either of these methods.

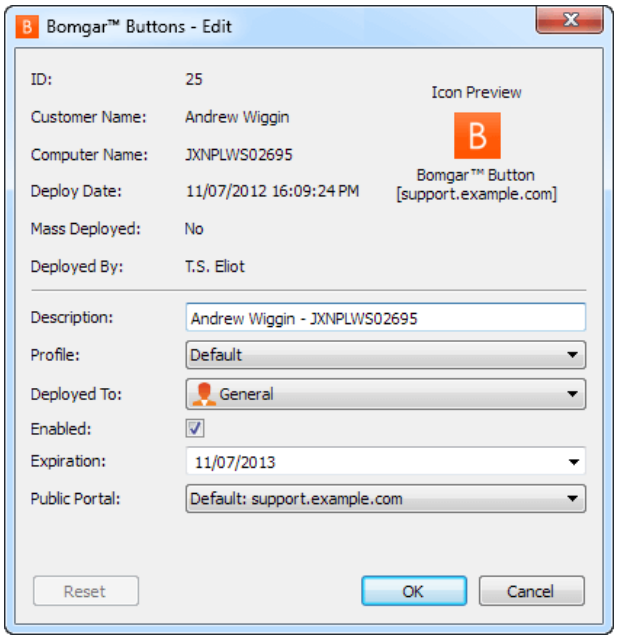
A session policy is assigned to a public portal from the **/login > Public Portals > Customer Client** page.

When a Jump Client is deployed within a support session, the Jump Client is by default associated with the same public portal as the support session. To change this, first customize the Jump Client and select a public portal from the properties dialog. All future sessions started from this Jump Client are associated with the selected public portal. Jump Clients deployed from the mass deployment wizard of the /login interface are also associated with a public portal, selected before deployment. You can change the public portal for a deployed Jump Client from the Jump interface of the representative console. View the Jump Client properties and select the public portal.



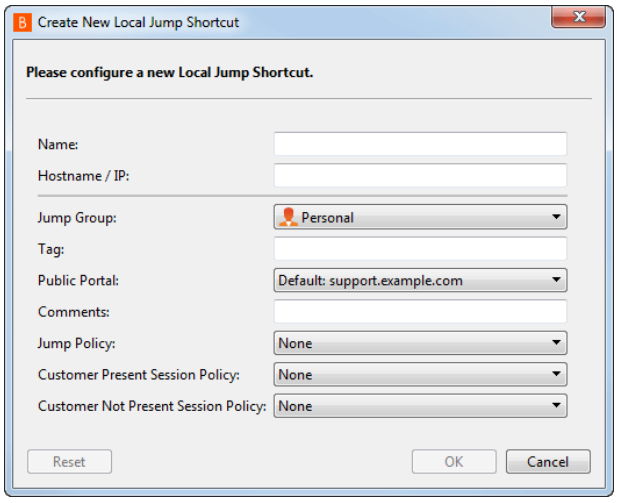
> ***Note:*** *According to the order of application, the Jump Client's session policy is applied before the public portal's session policy.*

Bomgar Buttons can also be associated with a public portal. When deployed within a support session, the Bomgar Button is by default associated with the same public portal as the support session. Bomgar Buttons deployed from the mass deployment wizard of the /login interface are also associated with a public portal, selected before deployment. You can change the public portal for a deployed Bomgar Button from the Bomgar Button management interface of the representative console. Edit a Bomgar Button and select the public portal.



> ***Note:*** *Session policies cannot be directly associated with specific Bomgar Buttons. However, based on the order of application, the session policy for the Bomgar Button's public portal is applied before any other session policies.*

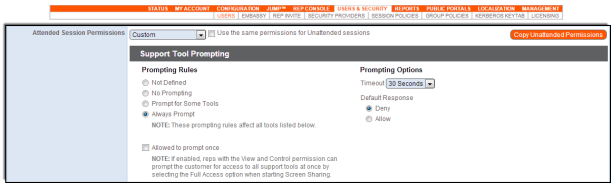
When creating a Jump Shortcut either from within the representative console or from /login, you can select a public portal to be associated with the shortcut. You can change the public portal for Jump Shortcut from the Jump interface of the representative console. View the Jump Shortcut properties and select the public portal.



> ***Note:*** *According to the order of application, a session policy applied to a Local Jump, Remote Jump, or Shell Jump takes precedence over the public portal's session policy.*

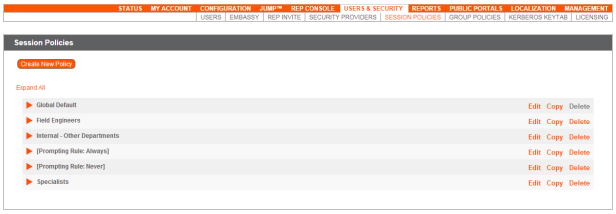
Ad-hoc Jump To sessions (started via Jumpoint or local push) are also affected, although all Jump To sessions are associated with the default public portal and cannot be changed. However, if you have assigned a session policy to your default public portal, that session policy is by association applied to Jump To sessions. This encompasses Local and Remote Jumps, Local and Remote RDP sessions, Local and Remote VNC sessions, Shell Jump sessions, and Intel® vPro sessions.

Representatives



A session policy can be applied to a representative by means of their user account, group policy, embassy, embassy user account, or rep invite profile. For all user objects other than rep invites, you can define separate policies for attended and unattended sessions run by this representative. Attended session policies apply to customer-initiated sessions and sessions started via Bomgar Button. Unattended session policies apply to sessions started through a Jump Client, Jumpoint, or local Jump. The rep invite profile is selected when the session owner creates the session invitation.

Global Default



Finally, the global default session policy must be defined. All permissions undefined by more highly ranked policies (i.e., Jump Item policies, public portal policies, representative policies) are set by the global default. Therefore, this policy cannot be deleted, and all permissions within this policy must be defined. Even if all of the policies applied to a session leave one or more permissions undefined, the global default policy ensures that those permissions are still defined for the session. This policy comes pre-defined with all permissions set to the lowest possible permission level. You may modify these settings as appropriate for your organization.

Applying Session Policies

When configuring a representative (user account, group policy, embassy, or embassy user), the administrator has three options for applying a session policy. When configuring a public portal or Jump Item, only the first two options are available. When assigning a rep invite profile, only the first option is available.

1. Use a previously-defined session policy.
    - From **/login > Users & Security > Session Policies**, session policies can be created independently and can then be applied to objects within BeyondTrust. You may not set individual permission options for the object to which a session policy is applied.
    - When applying a session policy to a representative within the /login interface (user account, group policy, embassy, or embassy user):
        - Selecting a session policy displays its description and a read-only view of its permissions.
        - If you wish to modify the selected policy, click **Edit**. This takes you to the **Session Policies** page, where you can edit the permissions set. Be aware that changing this policy changes the permissions for all objects that use this policy.
    - If a permission is undefined within the session policy, then the lower policies set that permission. If no other assigned policies set that option for the session, then the global default policy's rule is applied.
2. Do not associate a session policy.
    - Because no session policy is set for this object, the lower policies must define the permissions for the session.
    - If other session policies are defined but leave some permissions undefined, then the global default policy's rule is applied to those permissions.
    - If no other session policies are defined, the system falls back to the global default policy.

3. Define custom session permissions that are valid only for the object currently being configured.
    - No externally defined session policy applies. You set each permission individually.
    - If a permission is undefined, then the lower policies set that permission. If no other assigned policies set that option for the session, then the global default policy's rule is applied.
    - If no session policies are available to be assigned to user objects, then each permission must be defined.

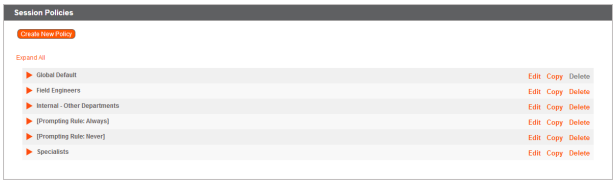
Furthermore, when configuring a representative, you can choose to define one policy for attended sessions and another for unattended sessions. You can also choose to apply the same permissions to both. Additionally, if configuring custom settings for both attended and unattended policies, you can copy the settings from one to the other.

Note: *Attended session policies apply to customer-initiated sessions and sessions started via Bomgar Button. Unattended session policies apply to sessions started through a Jump Client, Jumpoint, or local Jump.*

How Do I Set Up a Session Policy?

To create or edit a session policy, navigate to **/login > Users & Security > Session Policies**.

The **Session Policies** section lists available policies. Click the arrow by a policy name to quickly see where that policy is being used; its availability for users, rep invites, and Jump Items; the support tools configured; and the prompting configured. To create a session policy, click **Create New Policy**, or **Copy** an existing policy.



Create a unique name to help identify this policy. This name helps when assigning a session policy to users, public portals, and Jump Clients.Set a code name for integration purposes. If you do not set a code name, one is created automatically. Also give this policy a description to further detail the permissions available in this policy. The description is seen when applying a policy to user accounts, group policies, embassies, embassy users, and rep invites.

In the **Availability** section, choose if this policy should be available to assign to users (user accounts, embassies, and group policies). Also select if it should be available for representatives to use when inviting an external representative to join a session and if it should be available to assign to Jump Items. If this session policy is already in use, you should see the number of users, public portals, and Jump Items using this policy.

For all of the permissions that follow, you can choose to enable or disable the permission, or you can choose to set it to **Not Defined**. Session policies are applied to a session in a hierarchical manner, with Jump Clients taking the highest priority, then support portals, then users, and then the global default. If multiple policies apply to a session, then the policy with the highest priority will take precedence over the others. If, for example, the policy applied to a Jump Client defines a permission, then no other policies may change that permission for the session. To make a permission available for a lower policy to define, leave that permission set to **Not Defined**.

Set which support tools should be enabled or disabled with this policy, as well as which tools should prompt the customer for permission. Permissions and support tools are described in more detail in the Admin Interface at www.beyondtrust.com/docs/remote-support/getting-started/admin/session-policies.htm.

Click **Save Policy** to make this policy available.

Additionally, you can export a session policy from one site and import those permissions into a policy on another site. Edit the policy you wish to export and scroll to the bottom of the page. Click **Export Policy** and save the file.

You may now import those policy settings to any other BeyondTrust site that supports session policy import. Create a new session policy and scroll to the bottom of the page. Browse to the policy file and then click **Import Policy**. Once the policy file is uploaded, the page refreshes, allowing you to make modifications. Click **Save Policy** to make the policy available.

Best Practices

Policies for Specific Object Types

You may find it helpful to create session policies for specific types of objects. For example, create a session policy to be used only for certain Jump Clients and another only for a particular support portal. To help you remember which policies are created for which objects, preface the name of the session policy with the object type for which you have created the policy (e.g., "[Jump Client] Screen Sharing Only").

Only Essentials Defined

When defining session policies, set only the permissions you know are required for a given scenario. Be careful which permissions you allow, especially when defining policies for Jump Items or support portals. Remember that allowing a permission for a higher-ranking session policy means that permission is available in the session, even if the representative's account disallows that permission. This effectively grants the representative permission to perform an action they are not normally allowed to do.

The contrary also holds true. For example, if you deny a Jump Item a permission unnecessarily, then even a highly privileged representative is unable to perform that action.

Specifically when configuring Jump Item or support portal session policies, set only what you know needs to be set, and leave all other permissions undefined. This allows those permissions to fall through to the next level, where the next applied session policy can allow or deny those permissions as appropriate.

Lowest Privileges for Global Policy

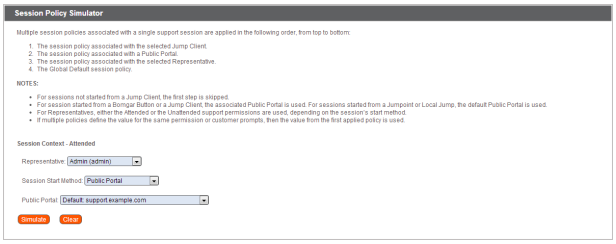
Ideally, a session's permissions should be set by its applied policies and should never reach the global default policy. However, because the global default session policy is the fallback policy for all sessions, set the permissions in this policy to the lowest privileges you would wish to allow. Any permissions which should be available for every representative in every possible session may be set to **Allow**. However, for any permissions which should be denied to some representatives or which should be disallowed for some end-users, set the policy to **Deny**.

How Do I Troubleshoot Session Policies?

Session policies can be applied to multiple objects within the same session. Because layering policies can be complex, you can use the **Session Policy Simulator** to determine what the outcome will be. Additionally, you could use the simulator to troubleshoot why a permission is not available when you expected it to be.

To access the session policy simulator, navigate to **/login > Users & Security > Session Policies**. Scroll to the bottom of the page.

Start by selecting the representative performing the session. This dropdown includes user accounts, embassy user accounts, and rep invite policies.



Next, select the session start method. This can be one of **Public Portal**, **Bomgar Button**, **Jumpoint, Local Jump, Jump Client**, or any Jump Item type.

If you selected **Public Portal**, choose the public portal to use for this simulation of a customer-initiated session.

If you selected **Bomgar Button**, search for a deployed Bomgar Button by profile, associated public portal, associated queue, computer name, or description. The associated public portal is automatically selected above.

Because local Jumps and Jumpoints are always associated with the default public portal, there are no further settings to define.

If you selected **Jump Client** or a Jump Shortcut, search for a pinned Jump Item by name, comments, Jump Group, tag, or associated public portal. The associated public portal is automatically selected above. You can choose whether the customer should appear as present or not.

Click **Simulate**. In the area below, the permissions configurable by session policy are displayed in read-only mode. You can see which permissions are allowed or denied as a result of the stacked policies, as well as which policy set each permission.

Session Policy Use Cases

Diverse Group of Clients

In this use case, your organization supports a diverse group of clients. Among your clients are a warehouse and a bank, which have very different security requirements. The primary support goal of the warehouse is for you to resolve their issues as quickly as possible, which means that they want you to have full control of their systems with only one prompt. On the other hand, the bank's primary goal is security, which means they want you to have view-only access and to prompt for each permission.

While you could assign specific representatives to handle each client, doing so would be an inefficient use of your support staff. It is more effective to allow your entire support staff to interact with all of your clients.

With session policies, a representative can handle sessions with permissions assigned from the support portal through which the session originated. Configure session permissions for the warehouse in one policy and permissions for the bank in another. Then apply the first policy to the warehouse portal and the second policy to the bank portal. Now, any session coming through the warehouse portal has the warehouse permissions applied, and any session coming from the bank portal has the bank permissions applied.

These portal-specific permissions override representative permissions. Any undefined permissions fall through to be defined first by representative permissions and then by the global default session policy.

Desktop and Server Jump Clients

In this use case, you have a number of Jump Clients installed on two different types of machines: servers and end-user computers. Company policy requires that end-users are always prompted before representatives can gain access. However, prompting prohibits access to servers because no end-user is there to respond to the prompt.

Configure two session policies: one that requires prompting and one that never prompts. Apply the first session policy to end-user Jump Clients and the second to server Jump Clients. Now, when a representative Jumps to an end-user system, the user is prompted for permission. When the same representative Jumps to a server, no prompting occurs.

These Jump Client-specific permissions override public portal permissions and representative permissions. Any undefined permissions fall through to be defined first by public portal permissions, then by representative permissions, and finally by the global default session policy.

Session Policy Examples

The table below contains some examples of valid configurations for session policies. These session policies are used in the examples that follow. Some session policies in the table are not used in the examples and are provided only as models of valid session policy configurations.

Name	Prompting				Support Tools			
	Which tools?	*Prompt Once*	*Timeout*	*Default*	*Screen Sharing Permissions*	*Screen Sharing Prompting*	*File Transfer Permission*	*File Transfer Prompting*
A	All	No	30 seconds	Deny	Allow	Always	Allow	Always
B	All	-	1 minute	Allow	Deny	Always	Allow	Always
C	All	-	15 seconds	Allow	Not Defined	Always	Not Defined	Always
D	All	-	30 seconds	Deny	Deny	Always	Deny	Always
E	Some	Yes	1 minute	Allow	Allow	Always	Deny	Not Defined
F	Some	No	15 seconds	Allow	Allow	Always	Not Defined	Not Defined
G	Some	Yes	20 seconds	Allow	Allow	Always	Not Defined	Not Defined
H	None	-	-	-	Not Defined	Never	Not Defined	Never
I	None	-	-	-	Allow	Never	Not Defined	Never
J	None	-	-	-	Allow	Never	Deny	Never
K	Not Defined	-	-	-	Deny	Not Defined	Deny	Not Defined
L	Not Defined	-	-	-	Allow	Not Defined	Not Defined	Not Defined
M	Not Defined	-	-	-	Allow	Not Defined	Allow	Not Defined

Remember that the order of application is hard-coded by BeyondTrust and cannot be changed, and higher priority policies cannot be overridden. The order in which policies are applied is:

1. Jump Item
2. Public Portal
3. Representative
4. Global Default

In the examples below, the order of policies in each table represents the hierarchy of the policies applied to a session. Therefore, for example, the first row of a table may serve as a public portal policy, while the second row serves as a representative policy.

Example 1: First Policy Defines Everything

Name	Prompting				Support Tools			
	Which tools?	*Prompt Once*	*Timeout*	*Default*	*Screen Sharing Permissions*	*Screen Sharing Prompting*	*File Transfer Permission*	*File Transfer Prompting*
A	All	No	30 seconds	Deny	Allow	Always	Allow	Always
B	All	-	1 minute	Allow	Deny	Always	Allow	Always
Final	**All**	**No**	**30 seconds**	**Deny**	**Allow**	**Always**	**Allow**	**Always**

Policy A defines every permission, so the final result is equivalent to **Policy A**.

Example 2: One Permission Undefined

Name	Prompting				Support Tools			
	Which tools?	*Prompt Once*	*Timeout*	*Default*	*Screen Sharing Permissions*	*Screen Sharing Prompting*	*File Transfer Permission*	*File Transfer Prompting*
E	Some	Yes	1 minute	Allow	Allow	Always	Deny	Not Defined
A	All	No	30 seconds	Deny	Allow	Always	Allow	Always
Final	**Some**	**Yes**	**1 minute**	**Allow**	**Allow**	**Always**	**Deny**	**Always**

Policy A's file transfer prompt behavior is used because **Policy E** did not define it.

Example 3: Two Permissions Undefined

Name	Prompting				Support Tools			
	Which tools?	*Prompt Once*	*Timeout*	*Default*	*Screen Sharing Permissions*	*Screen Sharing Prompting*	*File Transfer Permission*	*File Transfer Prompting*
F	Some	No	15 seconds	Allow	Allow	Always	Not Defined	Not Defined
D	All	-	30 seconds	Deny	Deny	Always	Deny	Always
Final	**Some**	**No**	**15 seconds**	**Allow**	**Allow**	**Always**	**Deny**	**Always**

1. **Policy F** does not define a file transfer permission, so **Policy D**'s rule is used.
2. **Policy F** does not define a file transfer prompt behavior, so **Policy D**'s rule is used.

Example 4: Three Layered Policies

Name	Prompting				Support Tools			
	Which tools?	*Prompt Once*	*Timeout*	*Default*	*Screen Sharing Permissions*	*Screen Sharing Prompting*	*File Transfer Permission*	*File Transfer Prompting*
M	Not Defined	-	-	-	Allow	Not Defined	Allow	Not Defined
G	Some	Yes	20 seconds	Allow	Allow	Always	Not Defined	Not Defined
A	All	No	30 seconds	Deny	Allow	Always	Allow	Always
Final	**Some**	**Yes**	**20 seconds**	**Allow**	**Allow**	**Always**	**Allow**	**Allow**

1. **Policy M** does not define prompting options, so **Policy G**'s rules are used.
2. **Policy M** allows screen sharing.
3. **Policy M** does not define a screen sharing prompt behavior, so **Policy G**'s rule is used.
4. **Policy M** allows file transfer.
5. Neither **Policy M** nor **Policy G** specifies the file transfer prompt behavior, so **Policy A**'s rule is used.

Group Policies and Session Policies

Session policies associated with a group policy follow the same rules as other settings in a group policy.

To configure session policies for a group policy, the group policy must either:

- Define the permission **Allowed to provide remote support** as enabled, or
- Not define the permission **Allowed to provide remote support**.
    - If a representative using this group policy has permission to provide remote support, then the configured session policies apply to that representative.
    - If a representative using this group policy does not have permission to provide remote support, then the configured session policies are irrelevant.

The following tables show the expected behavior when configuring session policies with group policies.

Group Policy	Session Policy	Defined?	Override?
G1	S1	X	X
G2	-	-	-
G3	S2	X	-
G4	S3	X	-

User	Group Policies	Final Session Policy	Why?
U1	G1	S1	From G1
U2	G1, G2	S1	From G1; G2 does not have a session policy defined
U3	G1, G2, G3	S2	From G3; G3 overrides G1
U4	G3, G4	S2	From G3; G4 cannot override G3
U5	G4	S3	From G4

Note that in the case of U3, the final session policy is S2 and not a combination of S1 and S2. Session policies are not combined based on the order of the group policies. Rather, they follow the same mode of application as other permissions in group policies. Thus, the highest priority, non-overridable group policy sets the session policy for that representative.

However, if other types of session policies are applied to a session (public portal session policies and Jump Item session policies), they may be combined with the representative's session policy and/or with each other during the support session.