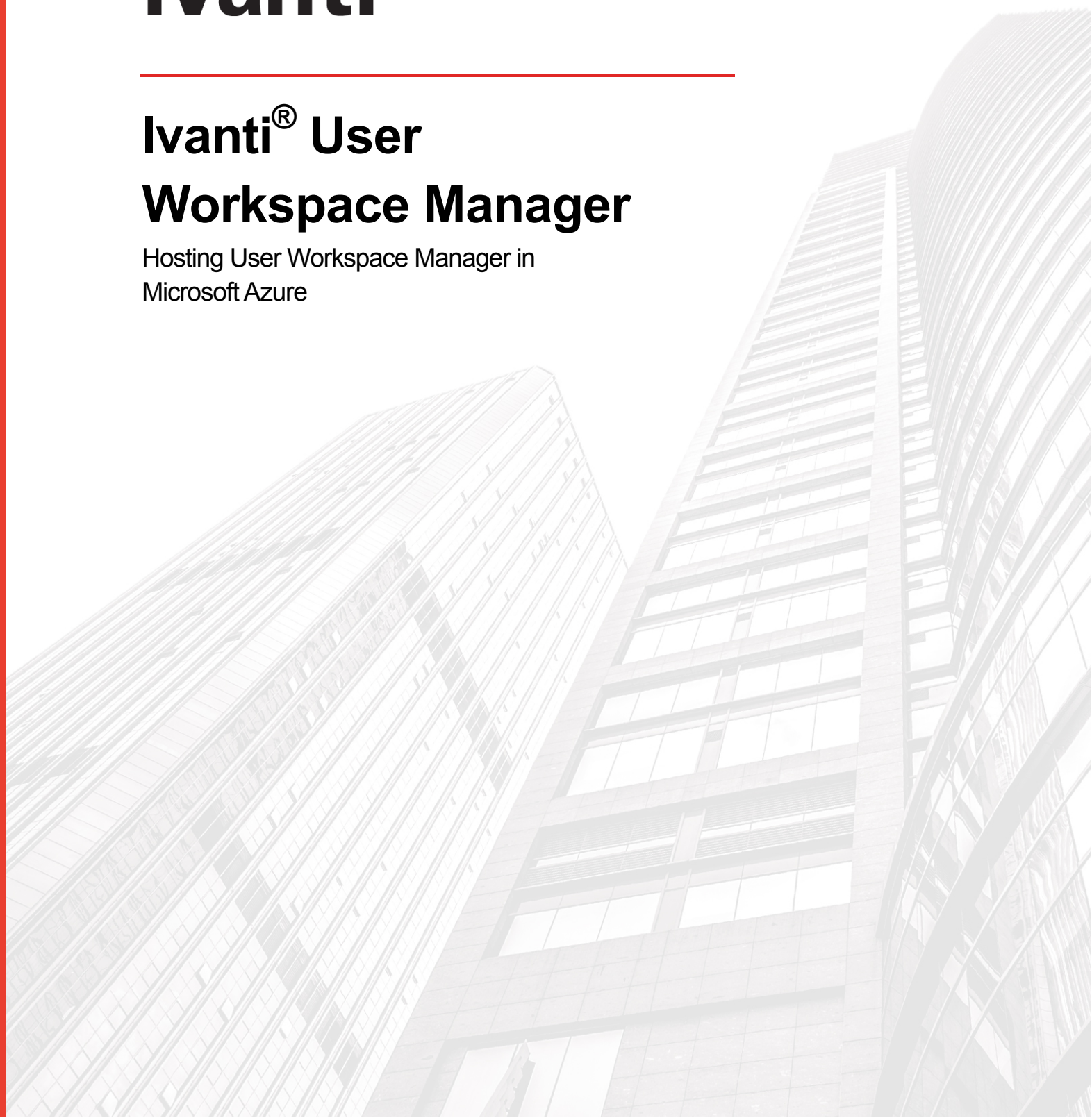
ivanti

Ivanti® User Workspace Manager

Hosting User Workspace Manager in Microsoft Azure



Contents

Purpose of this Document 4
Overview 4
Non-load balanced Azure Environment 4
Microsoft Azure Configuration 4
Microsoft Azure Virtual Machines 5
On-Premises Environment 6
Ivanti User Workspace Manager Configuration 6
Ivanti Management Server Configuration 7
Ivanti Personalization Server Configuration 7
Consoles 8
Overall Configuration 8
Basic Load Balanced Azure Environment 9
Microsoft Azure Configuration 9
Microsoft Azure Virtual Machines 9
Network Security Group 10
Virtual Network 11
Availability Set 12
Network Load Balancer 12
Storage Account 13
On-Premises Environment 14
Ivanti User Workspace Manager Configuration 14
Ivanti Management Server Configuration 15
Ivanti Personalization Server Configuration 15
Consoles 16
Overall Configuration 17

Contents (cont'd)

Advanced Load Balanced Azure Environment 18
Microsoft Azure Configuration 18
Microsoft Azure Virtual Machines 18
Network Security Group 21
Virtual Network 22
Availability Set 23
Network Load Balancer 23
Storage Account 25
On-Premises Environment 25
Ivanti User Workspace Manager Configuration 25
Ivanti Management Server Configuration 26
Ivanti Personalization Server Configuration 26
Consoles 27
Overall Configuration 28
Additional Reading 29

Purpose of this Document

The purpose of this document is to provide Ivanti customers and partners with a series of recommendations when working with Ivanti® User Workspace Manager and the Microsoft Azure Cloud Computing and Services platform.

It should be noted that this document will not include details on the installation or configuration of Ivanti User Workspace Manager or the Microsoft Azure platform.

Overview

The document serves to provide the reader an overview of configuring the Ivanti User Workspace Manager within the Microsoft Azure platform.

Only the Ivanti components applicable to this document are detailed and discussed. For full details of the Ivanti User Workspace Manager solution, consult the product documentation available at http://www.Ivanti.com.

Further details relating to the Microsoft Azure platform are available at https://azure.microsoft.com/en-gb/.

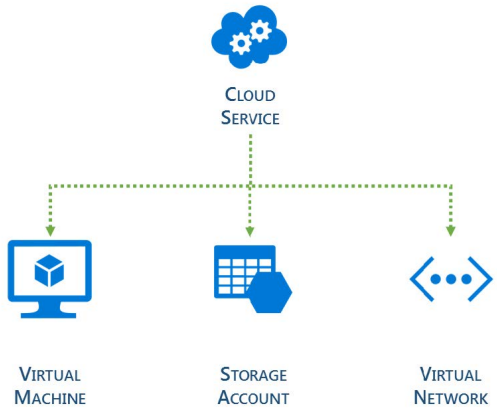
This document is composed of three sections:

- Hosting User Workspace Manager in a non-load balanced Azure environment
- Hosting User Workspace Manager in a basic load balanced Azure environment
- Hosting User Workspace Manager in an advanced load balanced Azure environment

Non-load balanced Azure Environment

Microsoft Azure Configuration

The Microsoft Azure Portal allows for the administration of everything from simple web-based apps to complex cloud applications from a single unified console. For this section of the document, the following resources were created via the Azure Portal.



Note: When using only a single Virtual Machine, it isn't necessary for a Virtual Network to be used. If, however, a separate Virtual Machine had been commissioned for the Microsoft SQL Server, then the Virtual Network would have been required for communication between both Virtual Machines.

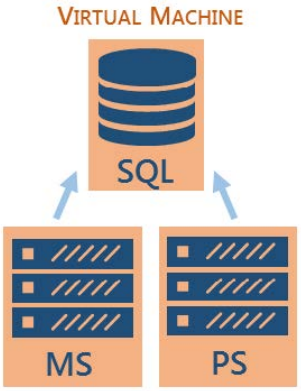
The configuration, when viewed from the Azure Portal, is shown below.

NAME	TYPE	RESOURCE GROUP	LOCATION
8kportalvhdsn03lmjmk4yg2	Storage accou...	Default-Storag...	West Europe
AppSenseMgtSvr	Cloud service (...	AppSenseMgt...	West Europe
AppSenseMgtSvr	Virtual machin...	AppSenseMgt...	West Europe

Note: Azure Storage, or AStorage, is accessible from anywhere in the world, from any type of application, whether it is running in the cloud, on a desktop, or on an on-premises server. For the purposes of this scenario, AStorage was used to provide access to installation media, including the Ivanti User Workspace Manager platform.

Microsoft Azure Virtual Machines

The Microsoft Azure platform provides a flexible environment for implementing a wide range of computing solutions. These machines can be accessed via a Remote Desktop (RDP) session in a similar way to that of an on-premises server.



MS = AppSense Management Server
PS = AppSense Personalization Server

A single Virtual Machine was created and configured as follows:

- Microsoft Windows Server 2012 R2 with the necessary IIS Roles installed
- Microsoft SQL Server 2014 Standard Edition with Service Pack 1
- Ivanti User Workspace Manager v10
- VPN Client

The use of a single Virtual Machine does not indicate that Microsoft SQL Server and Ivanti User Workspace Manager must be co-installed. This was merely a decision of simplicity rather than necessity.

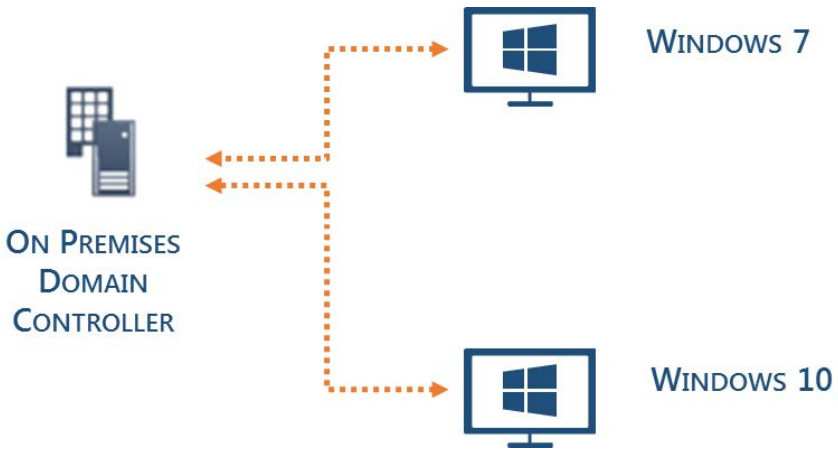
Azure has two options for hosting SQL Server workloads:

- Azure SQL Database: A SQL database that is native to the cloud – sometimes referred to as a Platform as a Service (PaaS) or Database or as a Service (DBaaS) – that is optimized for Software as a Service (SaaS) app development.
- SQL Server on Azure Virtual Machines: A SQL Server that is installed and hosted in the cloud on virtual machines, sometimes referred to as an Infrastructure as a Service (IaaS).

At the time of writing, it wasn't possible to use Ivanti User Workspace Manager and the Azure SQL Database method for hosting the Management and Personalization databases.

On-Premises Environment

An on-premises environment was built to prove that physical desktops can be managed from the cloud, for example, via an Ivanti® Environment Manager implementation housed within the Microsoft Azure platform. It's not intended to be a representation of a typical Ivanti customer implementation.



The on-premises environment consisted of:

- Microsoft Windows Server 2012 R2 configured as a Domain Controller
- Microsoft Windows 7 Ultimate
- Microsoft Windows 10 version 1607

In addition, a Virtual Private Network was configured to allow the Microsoft Azure hosted server to access and join the on-premises domain.

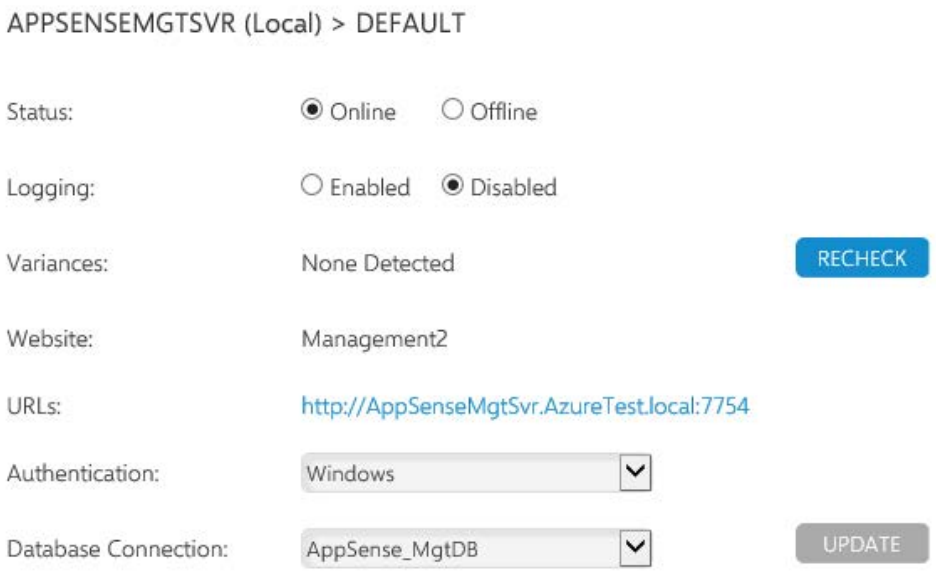
Ivanti User Workspace Manager Configuration

The Azure hosted Microsoft Windows Server 2012 R2 Virtual Machine was joined to the on-premises domain. User Workspace Manager v10 was installed using the Suite Installer, and the Server Configuration Portal was used to create the following databases within the workload of the SQL Server on an Azure Virtual Machine:

- Ivanti_MgtDB
- Ivanti_PersDB

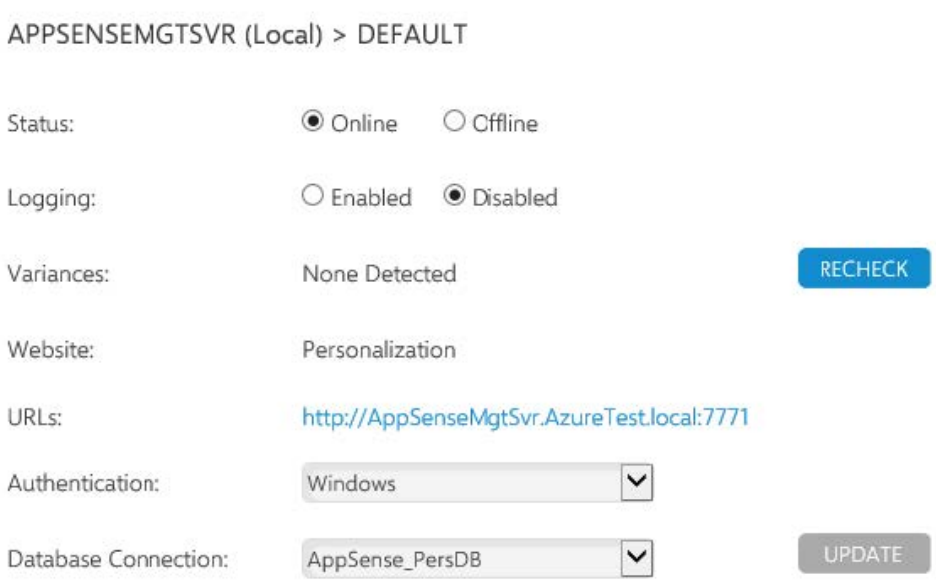
Ivanti Management Server Configuration

The Ivanti Management Server was configured in the following way:



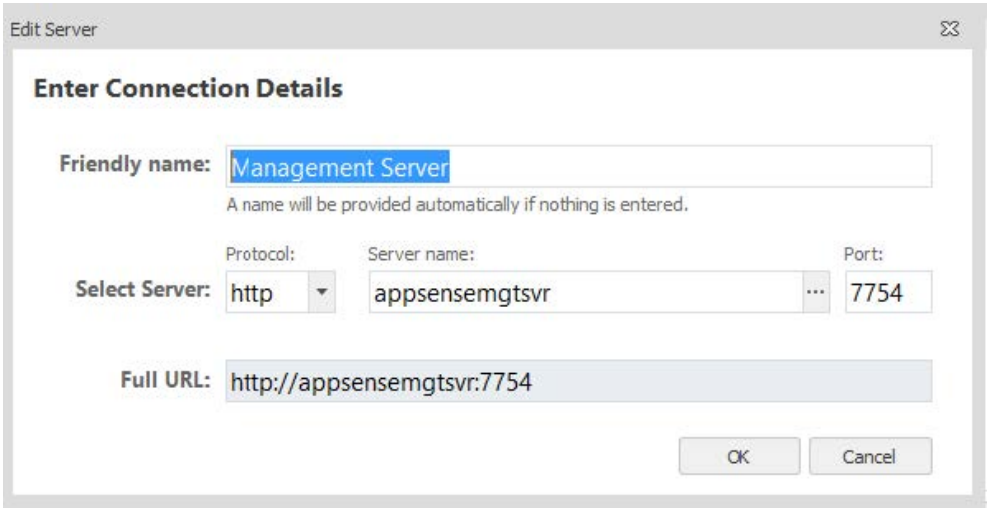
Ivanti Personalization Server Configuration

The Ivanti Personalization Server was configured in the following way:



Consoles

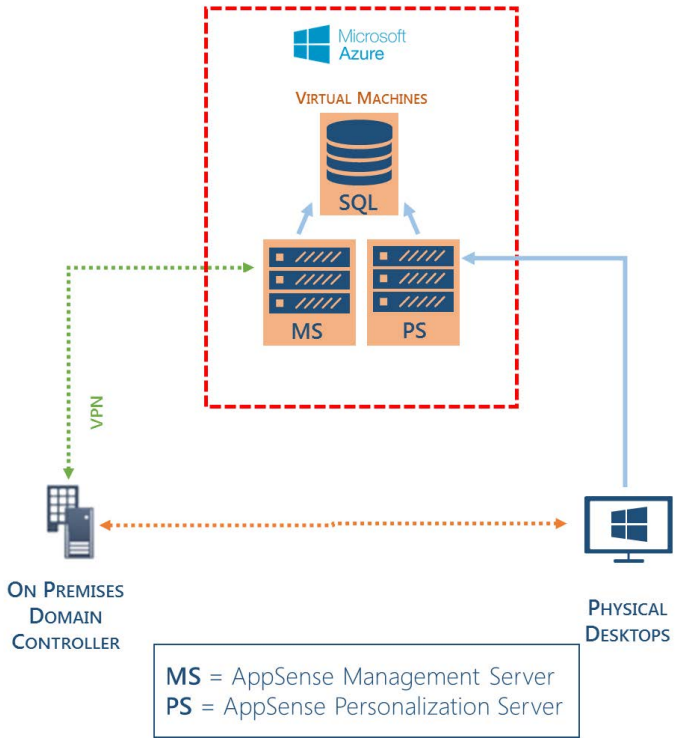
The Ivanti Management Center and Environment Manager consoles were configured to connect to the respective Azure-hosted servers. There was no bespoke configuration required.



Typical configuration such as Membership Rules and Access Credentials were configured and the agents for the Management Center, Application Control, Environment Manager, and Performance Manager were deployed. Again, no bespoke configuration was required.

Overall Configuration

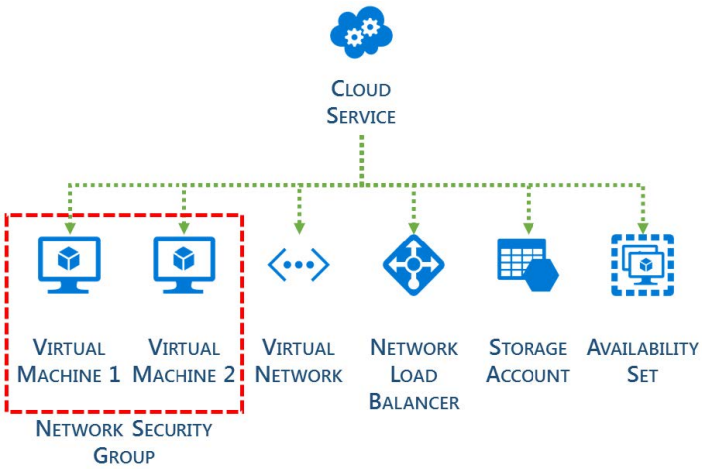
The diagram below provides an overview of the configuration of Ivanti User Workspace Manager and Microsoft Azure platform in a non-load balanced environment.



Basic Load-Balanced Azure Environment

Microsoft Azure Configuration

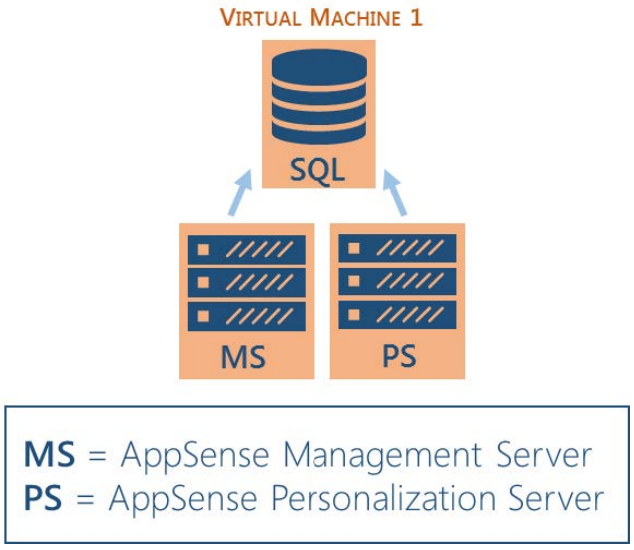
The following resources were created within the Azure Portal:



As can be seen from the diagram above, when compared to a non-load balanced environment, a number of additional Azure components are required.

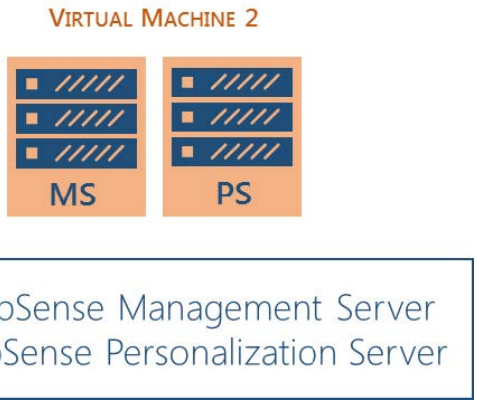
Microsoft Azure Virtual Machines

The Microsoft Azure platform provides a flexible environment allowing for a wide range of computing solutions to be implemented. These machines can be accessed via a Remote Desktop (RDP) session in a similar way to that of an on-premises server.



Virtual Machine 1 was created and configured as follows:

- Microsoft Windows Server 2012 R2 with the necessary IIS Roles installed
- Microsoft SQL Server 2014 Standard Edition with Service Pack 1
- Ivanti User Workspace Manager v10
- VPN Client



Virtual Machine 2 was created and configured as follows:

- Microsoft Windows Server 2012 R2 with the necessary IIS Roles installed
- Ivanti User Workspace Manager v10
- VPN Client

Network Security Group

A Network Security Group (NSG) contains a list of Access Control List (ACL) rules that allow or deny network traffic to your Virtual Machine instances within a given Virtual Network. An NSG can be associated with a subnet – resulting in all ACL rules being applied to all VM instances in that subnet – or to an individual Virtual Machine.

In this instance, a single NSG was created to be used on the Virtual Network subnet:

Resource group
Default-Storage-WestEurope

Location
West Europe

Subscription name
Visual Studio Professional

Subscription ID
998922b7-1bfe-4059-9e10-bd105b34f2e5

Security rules
3 inbound, 0 outbound

Associated with
1 subnets, 1 network interfaces

3 Inbound security rules

PRIORITY	NAME	SOURCE	DESTINATION	SERVICE	ACTION
100	Internet	Internet	Any	Custom (Any/Any)	Allow
101	NLB	AzureLoa...	Any	Custom (Any/Any)	Allow
1000	default-allow-rdp	Any	Any	RDP (TCP/3389)	Allow

0 Outbound security rules

PRIORITY	NAME	SOURCE	DESTINATION	SERVICE	ACTION
No results.					

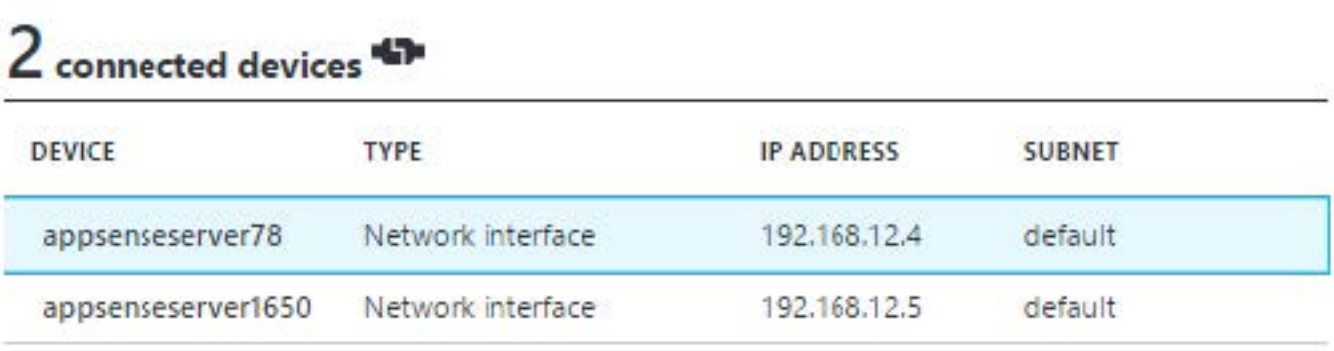
Virtual Network

For the Virtual Machines to be able to communicate successfully, an Azure Virtual Network is required. To allow for the Azure and on-premises networks to function, an address space was configured within the Azure portal in-line with the on-premises address space:

Similarly, a subnet was configured:

NAME	ADDRESS RANGE	AVAILABLE ADDR...	SECURITY GROUP	
default	192.168.12.0/24	249	-	...

When the Virtual Machines were created, they were added to the Azure Virtual Network:



DEVICE	TYPE	IP ADDRESS	SUBNET
appsenseserver78	Network interface	192.168.12.4	default
appsenseserver1650	Network interface	192.168.12.5	default

The Azure portal offers a diagrammatical view of a Virtual Network; the diagram below shows the Virtual Network that was created in this instance:

Availability Set

When working with two or more Virtual Machines within the Azure platform you should use an Availability Set for each application tier. As an example, you might place domain controllers in one Availability Set, SQL Servers in a second Set, and Web Servers in a third. Without this grouping, Azure is unable to distinguish between the application tiers for each Virtual Machine.

This could lead to a single point of failure in the hardware infrastructure, causing an outage or a planned maintenance event rebooting all Virtual Machines in the same application tier simultaneously.

The two Virtual Machines that are used in this configuration were added to an Availability Set.

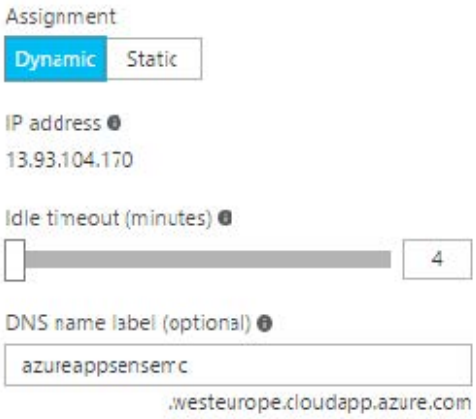
Network Load Balancer

The Azure Load Balancer provides high availability and network performance for the applications it serves. Layer 4 (TCP, UDP) is employed to distribute incoming traffic amongst all healthy instances of services that have been defined in a load-balanced set.

To configure a load balancer within the Azure platform, a Health Probe must be created. This is used to assess the health of an instance that is being served by the load balancer:

Once a Health Probe has been configured, the Backend Pool that will be used by the load balancer when distributing incoming traffic is created. The Availability Set and the relevant Virtual Machines are included to create the Backend Pool:

A Front-end IP Pool is used to associate a public IP address or addresses with the load balancer. These addresses are created from within the Azure portal and can also be associated with a DNS name:



The final configuration task is to create the actual Load balancing rules that will be used to distribute the incoming traffic. For the purposes of this document, the following rules have been created within the load balancer configuration:

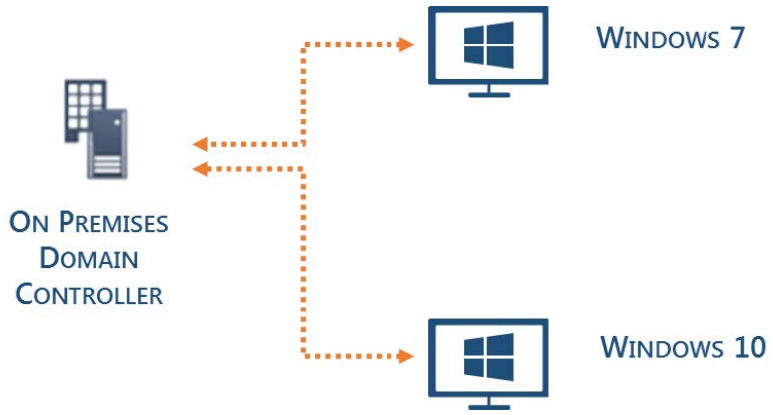
NAME	LOAD BALANCING RULE	BACKEND POOL	PROBE
FrontEndPort80	FrontEndPort80 (TCP/80)	ManagementServers	HealthProbe
ManagementServer	ManagementServer (TCP/7751)	ManagementServers	HealthProbe
PersonalizationServer	PersonalizationServer (TCP/7771)	PersonalizationServers	HealthProbe

Storage Account

Azure Storage was configured to allow installation media such as Ivanti User Workspace Manager to be made available to all Virtual Machines.

On-Premises Environment

An on-premises environment was built to prove that physical desktops can be managed from the cloud, for example, via an Environment Manager implementation housed within the Microsoft Azure platform.



The on-premises environment consists of:

- Microsoft Windows Server 2012 R2 configured as a Domain Controller
- Microsoft Windows 7 Ultimate
- Microsoft Windows 10 version 1607

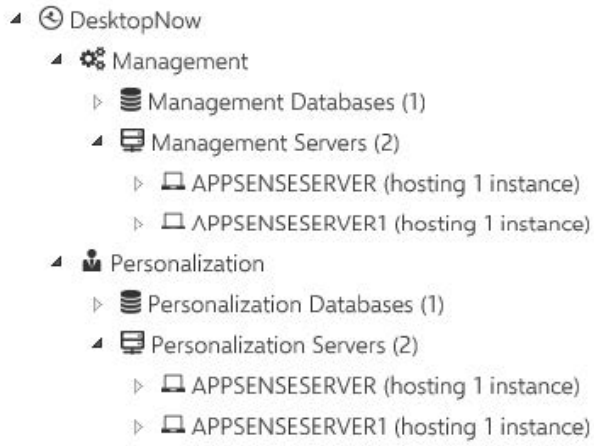
In addition, a Virtual Private Network was configured to allow the Microsoft Azure-hosted servers to join and access the on-premises Domain.

Ivanti User Workspace Manager Configuration

The Azure-hosted Microsoft Windows Server 2012 R2 Virtual Machines were joined to the on-premises domain. User Workspace Manager v10 was installed using the Suite Installer on both of the Virtual Machines. Finally, the Server Configuration Portal was used to create the following databases within the workload of the SQL Server on one of the Azure Virtual Machines:

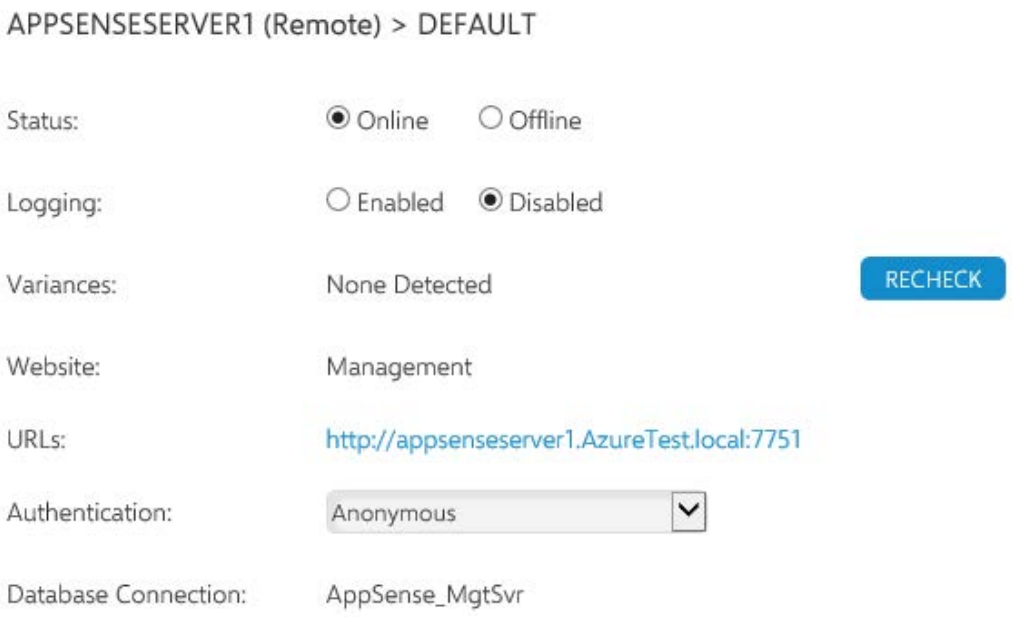
- Ivanti_MgtDB
- Ivanti_PersDB

Each Virtual Machine was then configured to host an instance of the Ivanti Management and Personalization Server:



Ivanti Management Server Configuration

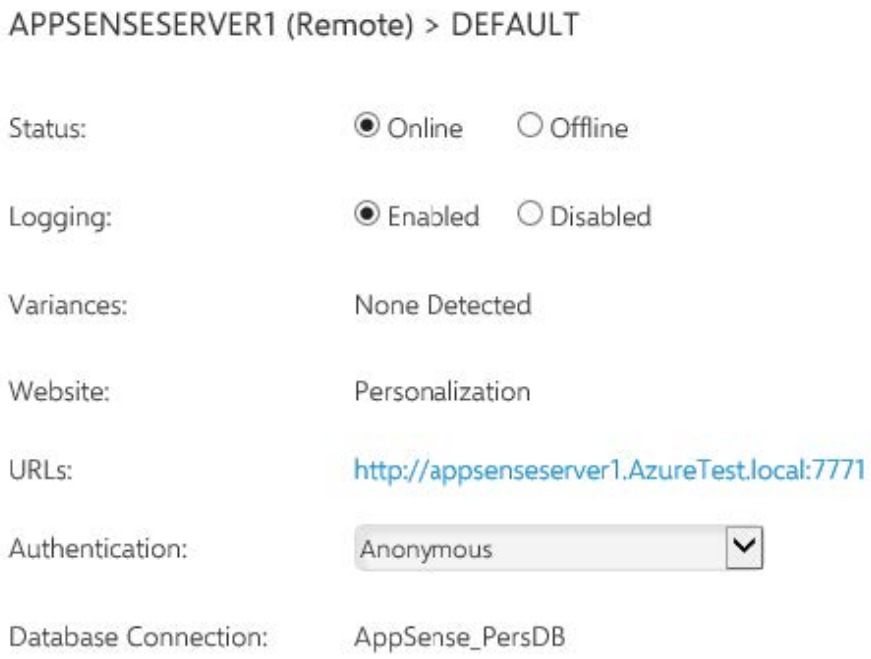
Both Management Servers were configured in the following way:



Note: When using a load balanced configuration, it is necessary to set the Authentication method to Anonymous.

Ivanti Personalization Server Configuration

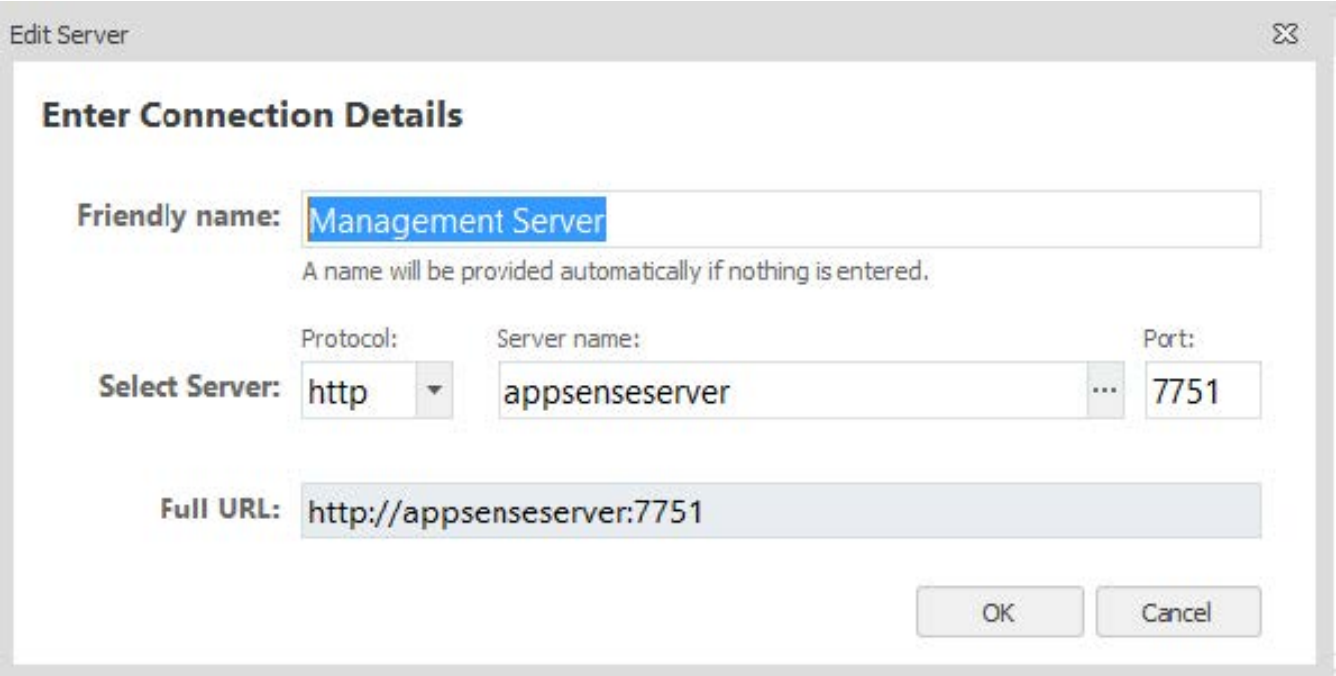
Both Personalization Servers were configured in the following way:



Note: When using a load balanced configuration, it is necessary to set the Authentication method to Anonymous.

Consoles

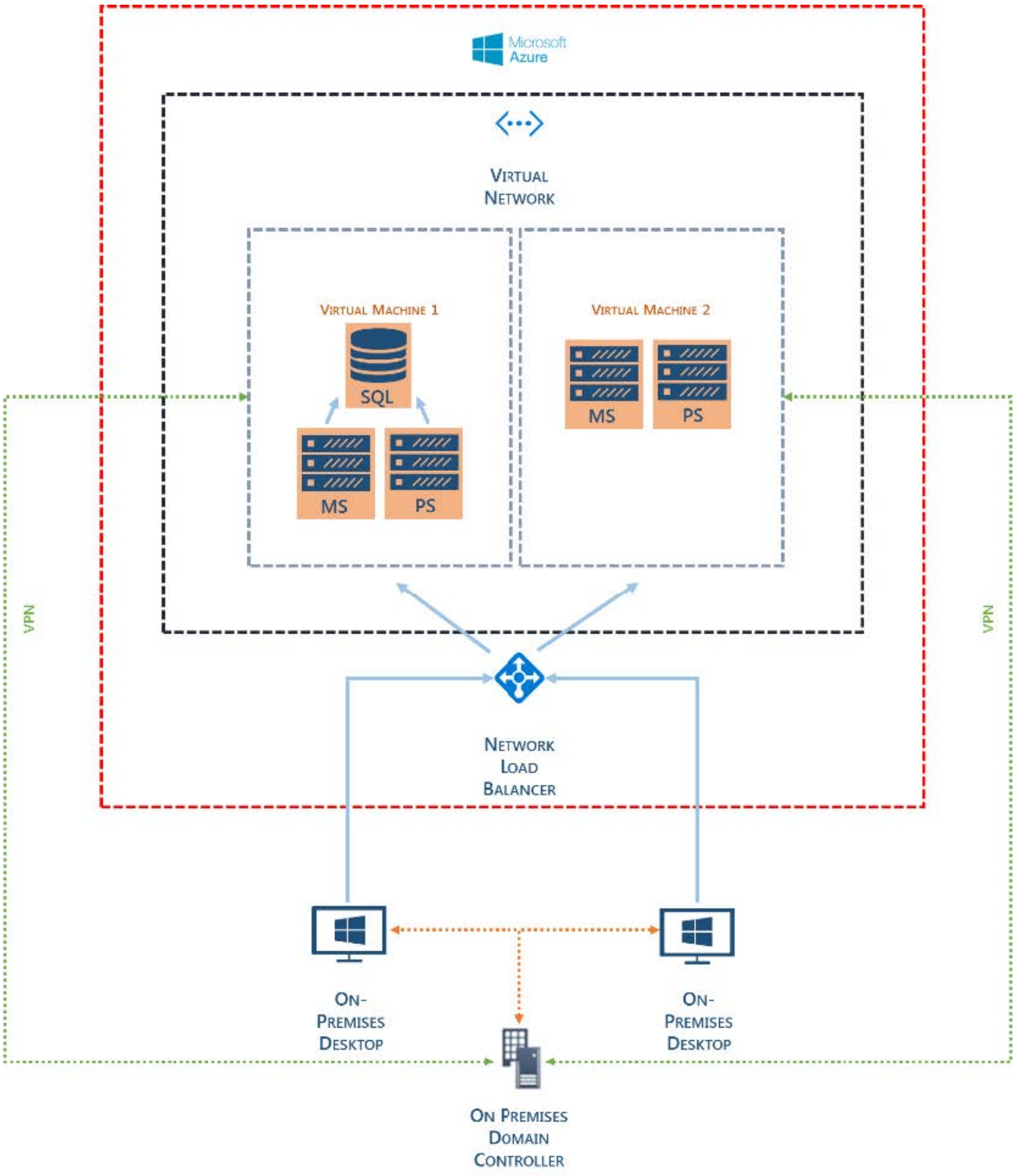
The Ivanti Management Center and Environment Manager consoles were configured to connect to the respective Azure-hosted servers. There was no bespoke configuration required:



Typical configuration such as Membership Rules and Access Credentials were configured and the agents for the Ivanti Management Center, Application Control, Environment Manager, and Performance Manager were deployed. Again, no bespoke configuration was required.

Overall Configuration

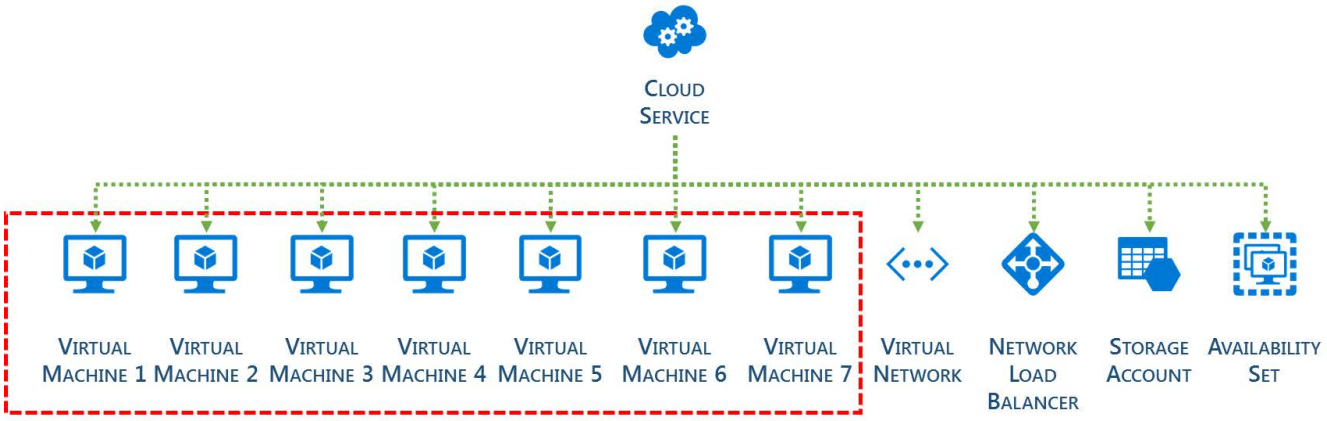
The diagram below provides an overview of the configuration of Ivanti User Workspace Manager and Microsoft Azure platform in a load balanced environment:



Advanced Load-Balanced Azure Environment

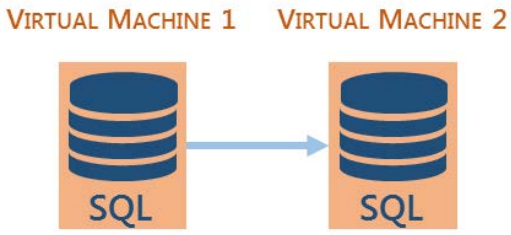
Microsoft Azure Configuration

The following resources were created within the Azure Portal:



Microsoft Azure Virtual Machines

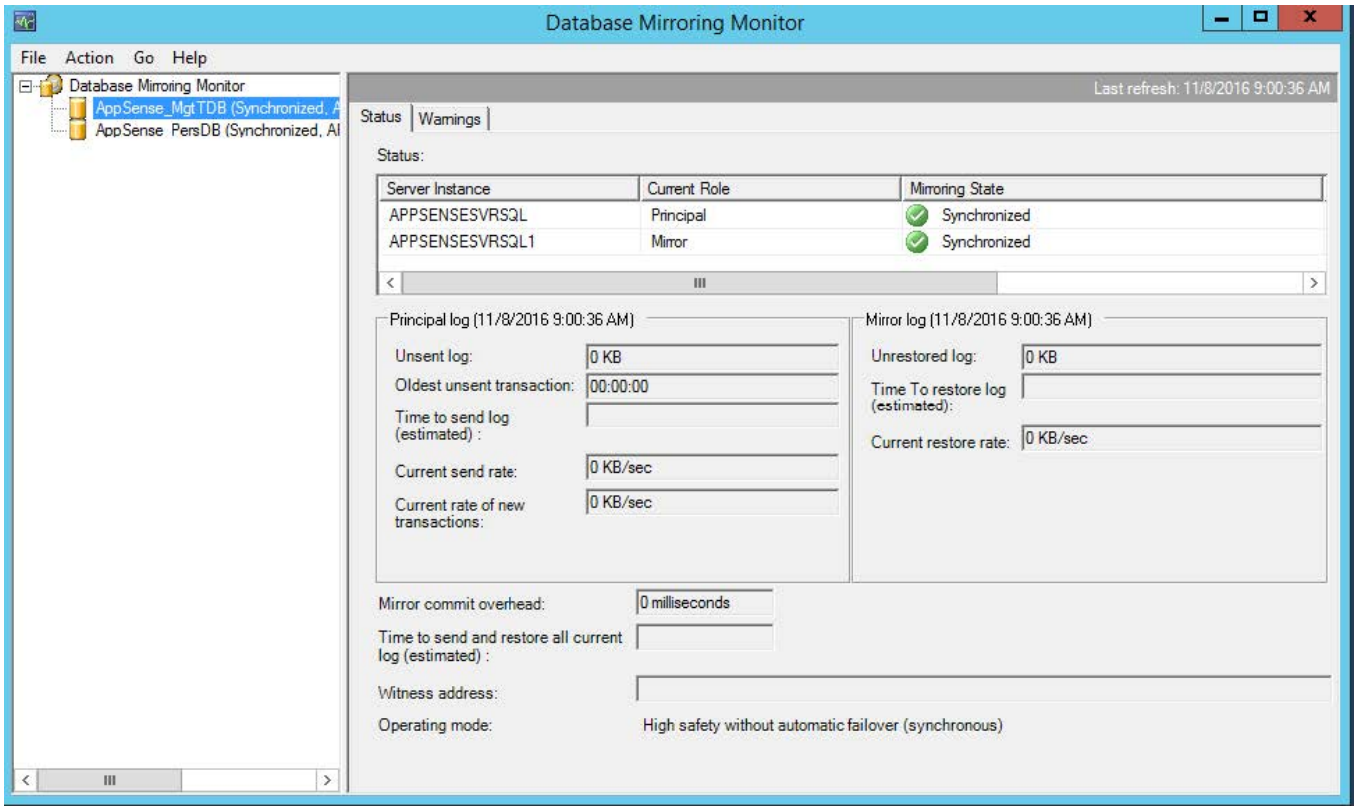
The Microsoft Azure platform provides a flexible environment allowing for a wide range of computing solutions to be implemented. These machines can be accessed via a Remote Desktop (RDP) session in a similar way to that of an on-premises server.

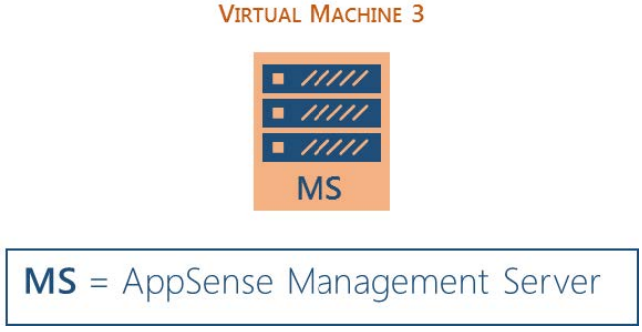


Virtual Machines 1 and 2 were created as follows:

- Microsoft Windows Server 2012 R2
- Microsoft SQL Server 2014 Standard Edition with Service Pack 1 with Database Mirroring enabled
- VPN Client

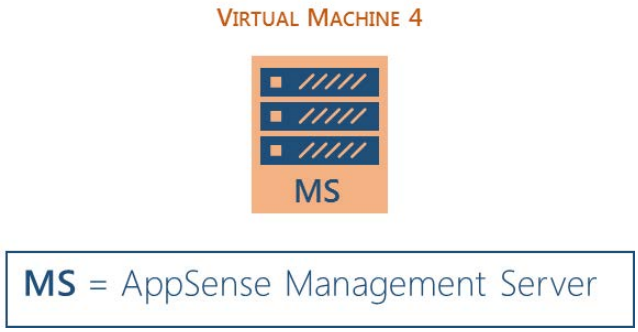
Database Mirroring was configured using Microsoft Best Practices. The following Microsoft TechNet article can be used as a starting point. The following illustrates the high-level configuration:





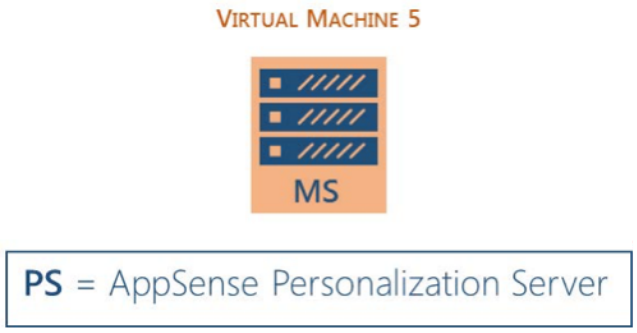
Virtual Machine 3 was created and configured as follows:

- Microsoft Windows Server 2012 R2 with the necessary IIS Roles installed
- Ivanti Management Server
- VPN Client



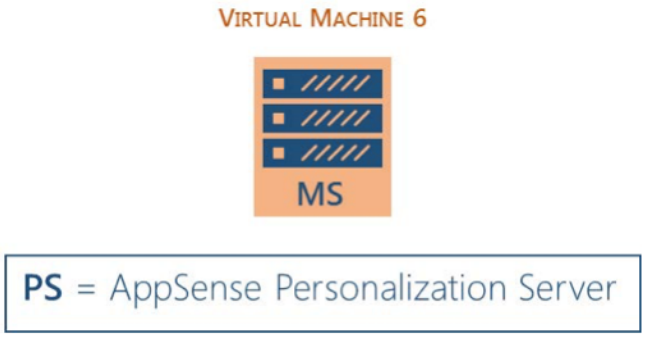
Virtual Machine 4 was created and configured as follows:

- Microsoft Windows Server 2012 R2 with the necessary IIS Roles installed
- Ivanti Management Server
- VPN Client



Virtual Machine 5 was created and configured as follows:

- Microsoft Windows Server 2012 R2 with the necessary IIS Roles installed
- Ivanti Management Server
- VPN Client



Virtual Machine 6 was created and configured as follows:

- Microsoft Windows Server 2012 R2 with the necessary IIS Roles installed
- Ivanti Management Server
- VPN Client

VIRTUAL MACHINE 7

Virtual Machine 7 was created and configured as follows:

- Microsoft Windows 10 version 1607
- VPN Client

Network Security Group

A Network Security Group (NSG) contains a list of Access Control List (ACL) rules that allow or deny network traffic to your Virtual Machine instances within a given Virtual Network. An NSG can be associated with a subnet – resulting in all ACL rules being applied to all VM instances in that subnet – or to an individual Virtual Machine.

In this instance, a single NSG was created to be used on the Virtual Network subnet:

Resource group
Default-Storage-WestEurope

Location
West Europe

Security rules
5 inbound, 1 outbound

Associated with
0 subnets, 6 network interfaces

5 Inbound security rules

PRIORITY	NAME	SOURCE	DESTINATION	SERVICE	ACTION
100	Internet	Internet	Any	Custom (Any/Any)	Allow
101	NLB	AzureLoa...	Any	Custom (Any/Any)	Allow
102	PersOPS	Any	Any	HTTP (TCP/80)	Allow
103	PersonalizationSer...	Any	Any	Custom (Any/7771)	Allow
1000	default-allow-rdp	Any	Any	RDP (TCP/3389)	Allow

Virtual Network

For the Virtual Machines to be able to communicate with each other, an Azure Virtual Network is required. To allow for the Azure and on-premises networks to function, an address space was configured within the Azure portal, in-line with the on-premises address space:

Similarly, a subnet was configured:

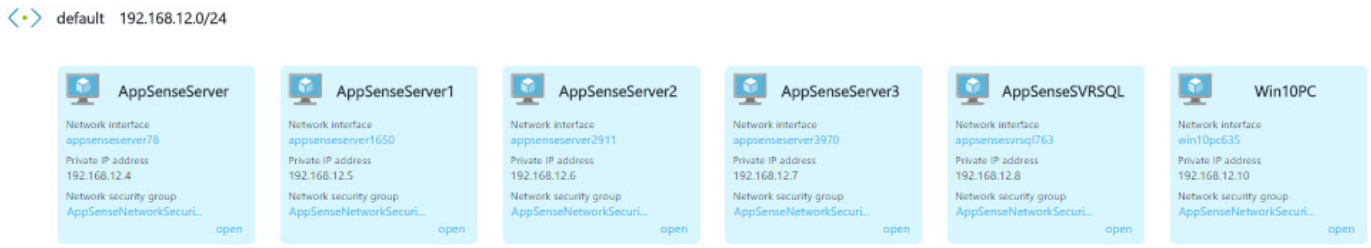
NAME	ADDRESS RANGE	AVAILABLE ADDR...	SECURITY GROUP	
default	192.168.12.0/24	249	-	...

When the Virtual Machines were created, they were added to the Azure Virtual Network:

7 connected devices

DEVICE	TYPE	IP ADDRESS	SUBNET
appsenseserver78	Network interface	192.168.12.4	default
appsenseserver1650	Network interface	192.168.12.5	default
appsenseserver2911	Network interface	192.168.12.6	default
appsenseserver3970	Network interface	192.168.12.7	default
appsensesvrsql763	Network interface	192.168.12.8	default
AppSenseMCNLB	Load balancer	192.168.12.9	default
win10pc635	Network interface	192.168.12.10	default

The Azure portal offers a diagrammatical view of a Virtual Network; the diagram below shows the Virtual Network that was created in this instance:



Availability Set

When working with two or more Virtual Machines within the Azure platform you should use an Availability Set for each application tier. As an example, you might place domain controllers in one Availability Set, SQL Servers in a second Set, and Web Servers in a third. Without this grouping, Azure is unable to distinguish between the application tiers for each Virtual Machine.

This could lead to a single point of failure in the hardware infrastructure, causing an outage or a planned maintenance event rebooting all Virtual Machines in the same application tier simultaneously.

The Virtual Machines that are used in this configuration were added to an Availability Set.

Network Load Balancer

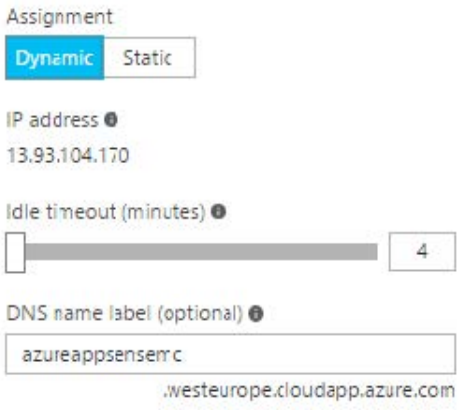
The Azure Load Balancer provides high availability and network performance for the applications it serves. Layer 4 (TCP, UDP) is utilized to distribute incoming traffic amongst all healthy instances of services that have been defined in a load-balanced set.

To configure a load balancer within the Azure platform, a Health Probe must be created. This is used to assess the health of an instance that is being served by the load balancer:

Once a Health Probe has been configured, the Backend Pool that will be used by the load balancer when distributing incoming traffic is created. The Availability Set and the relevant Virtual Machines are included to create the Backend Pool:

VIRTUAL MACHINE	STATUS	NETWORK INTERFACE	PRIVATE IP ADDRESS
▾ AllAppSenseServers (4 virtual machines)			
AppSenseServer	Running	appsenseserver78	192.168.12.4
AppSenseServer1	Running	appsenseserver1650	192.168.12.5
AppSenseServer2	Running	appsenseserver2911	192.168.12.6
AppSenseServer3	Running	appsenseserver3970	192.168.12.7
▾ BackendPool (2 virtual machines)			
AppSenseServer	Running	appsenseserver78	192.168.12.4
AppSenseServer1	Running	appsenseserver1650	192.168.12.5
▾ ManagementServers (2 virtual machines)			
AppSenseServer	Running	appsenseserver78	192.168.12.4
AppSenseServer1	Running	appsenseserver1650	192.168.12.5
▾ PersonalizationServers (2 virtual machines)			
AppSenseServer2	Running	appsenseserver2911	192.168.12.6
AppSenseServer3	Running	appsenseserver3970	192.168.12.7

A Frontend IP Pool is used to associate a public IP address or addresses with the load balancer. These addresses are created from within the Azure portal and can also be associated with a DNS name:



The final configuration task is to create the actual Load balancing rules that will be used to distribute the incoming traffic. For the purposes of this document the following rules have been created within the load balancer configuration:

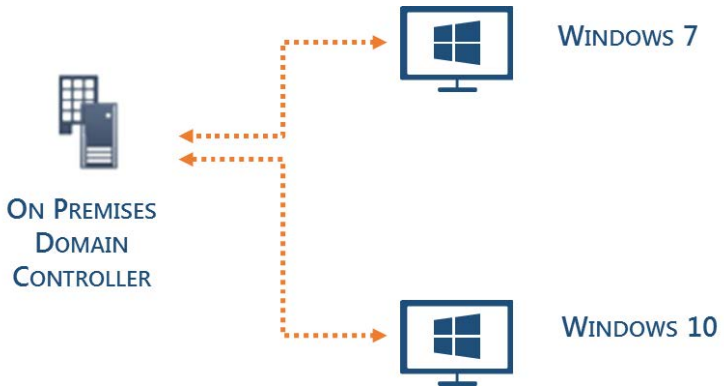
NAME	LOAD BALANCING RULE	BACKEND POOL	PROBE
FrontEndPort80	FrontEndPort80 (TCP/80)	ManagementServers	HealthProbe
ManagementServer	ManagementServer (TCP/7751)	ManagementServers	HealthProbe
PersonalizationServer	PersonalizationServer (TCP/7771)	PersonalizationServers	HealthProbe

Storage Account

Azure Storage was configured to allow installation media such as Ivanti User Workspace Manager to be made available to all Virtual Machines.

On-Premises Environment

An on-premises environment was built to prove that physical desktops can be managed from the cloud, for example via an Environment Manager implementation housed within the Microsoft Azure platform:



The on-premises environment consists of:

- Microsoft Windows Server 2012 R2 configured as a Domain Controller
- Microsoft Windows 7 Ultimate
- Microsoft Windows 10 version 1607

In addition, a Virtual Private Network was configured to allow the Microsoft Azure-hosted Virtual Machines to join and access the on-premises domain.

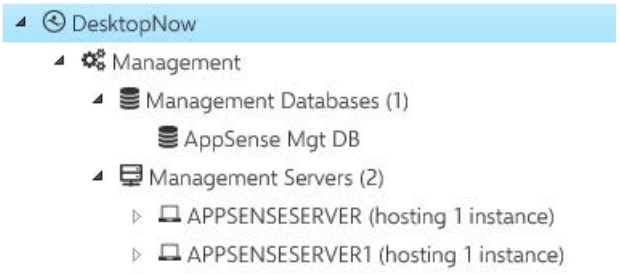
Ivanti User Workspace Manager Configuration

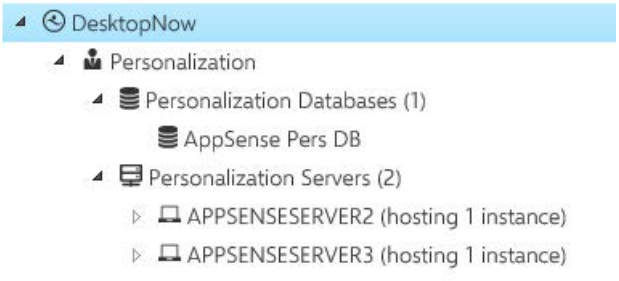
The Azure-hosted Microsoft Windows Server 2012 R2 Virtual Machines were joined to the on-premises domain. User Workspace Manager v10 was installed using the Suite Installer on each of the Ivanti Virtual Machines. Finally, the Server Configuration Portal was used to create the following databases within the workload of the SQL Server:

- Ivanti_MgtDB
- Ivanti_PersDB

Note: Both of these databases were configured to use Database Mirroring.

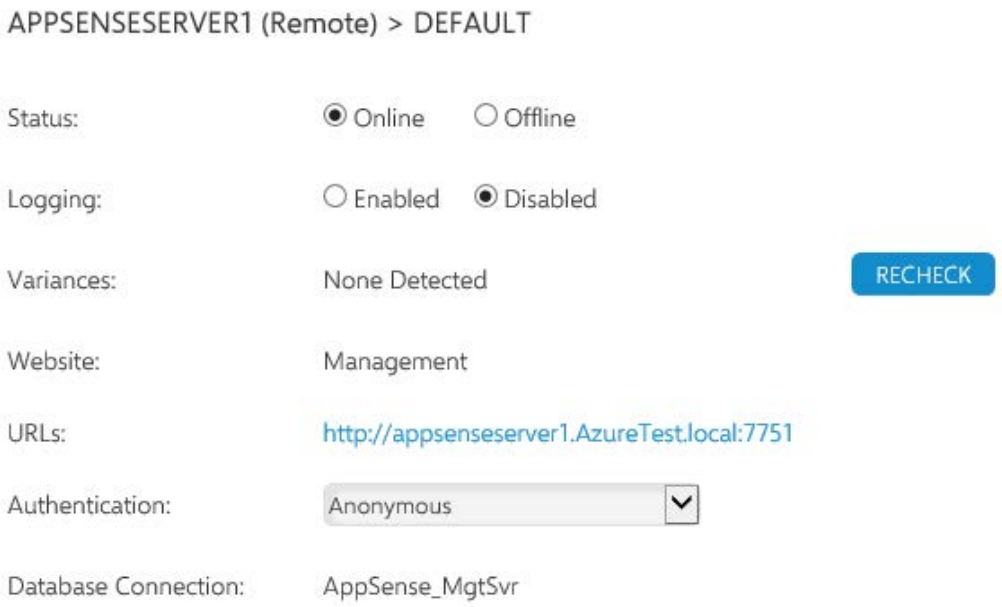
Each Virtual Machine was then configured to host either an instance of the Ivanti Management or Personalization Server:





Ivanti Management Server Configuration

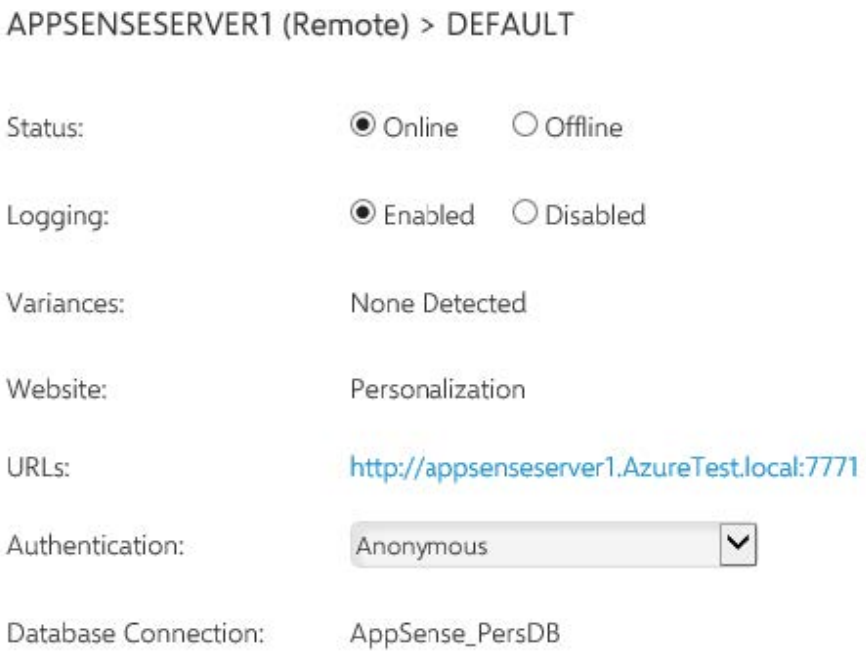
Both Management Servers were configured in the following way:



Note: When using a load balanced configuration, it was necessary to set the Authentication method to Anonymous

Ivanti Personalization Server Configuration

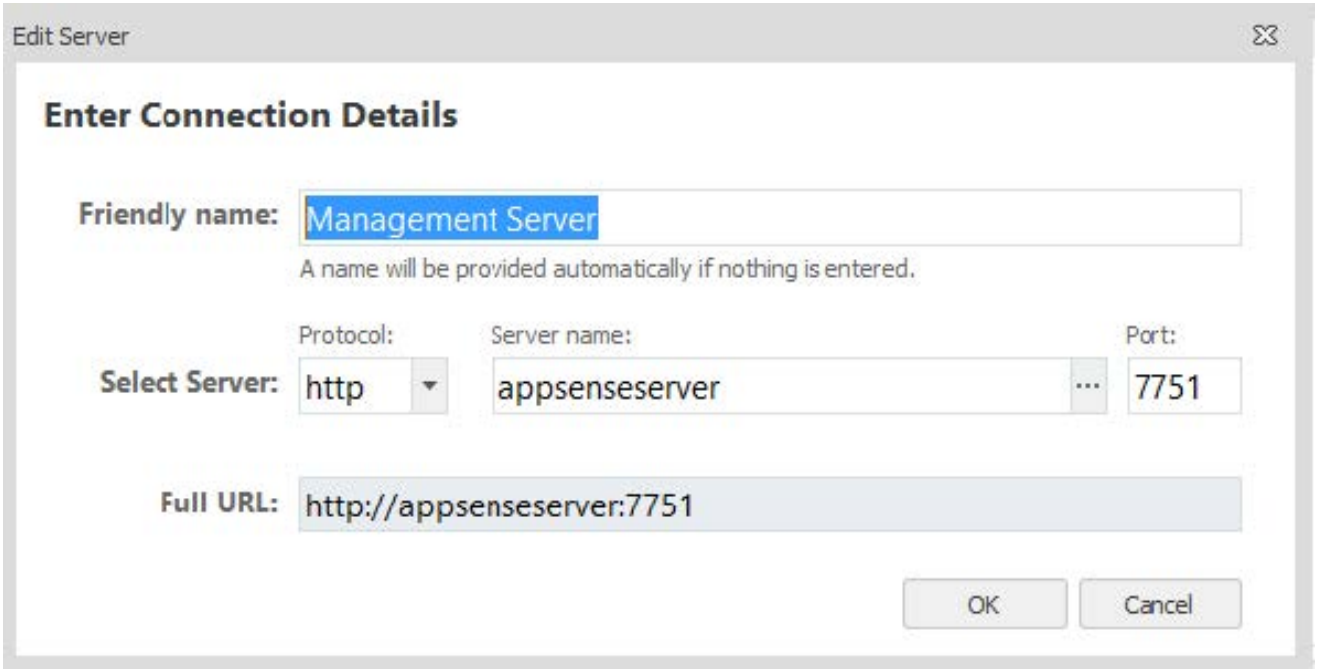
Both Personalization Servers were configured in the following way:



Note: When using a load balanced configuration, it was necessary to set the Authentication method to Anonymous.

Consoles

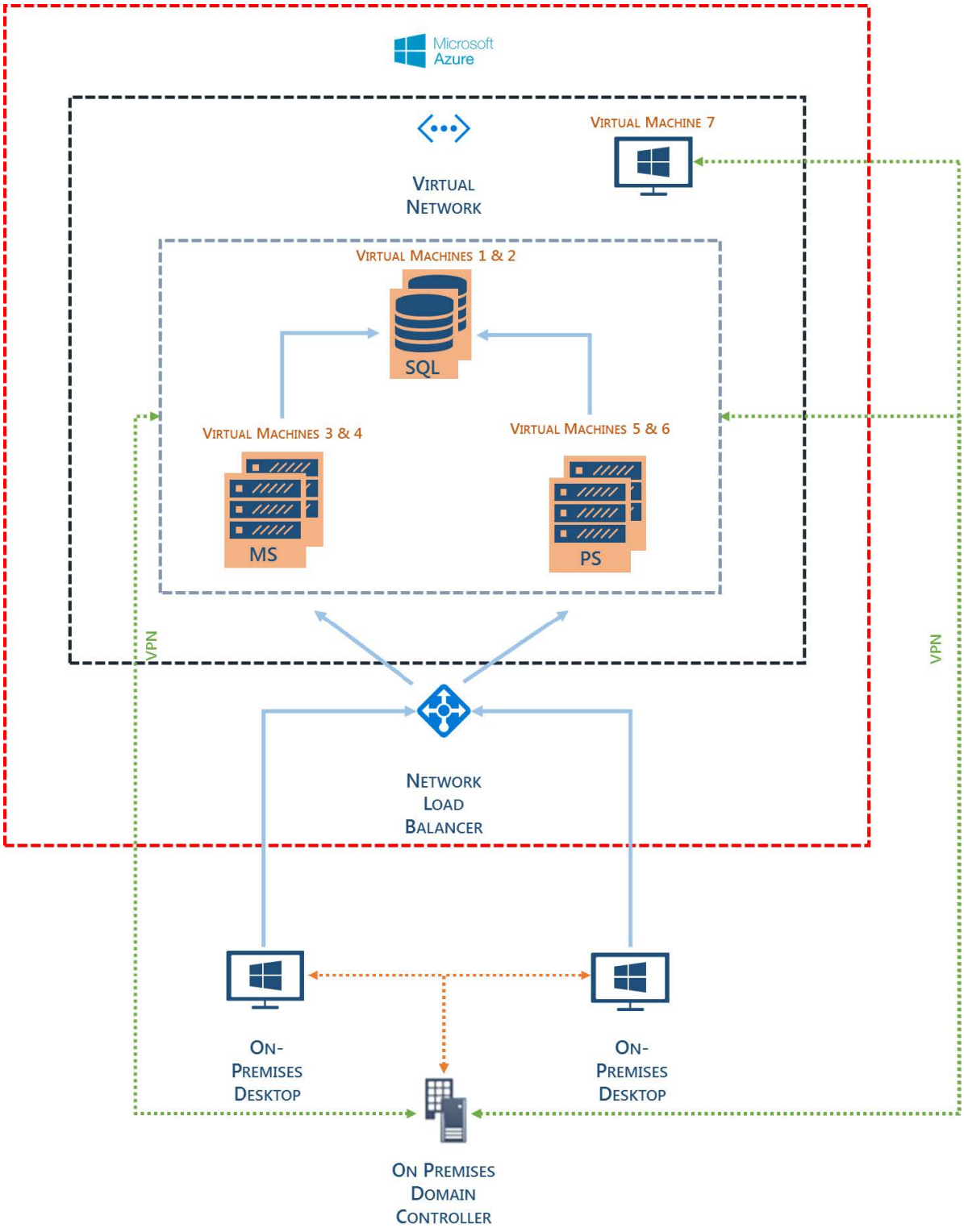
The Ivanti Management Center and Environment Manager consoles were configured to connect to the respective Azure-hosted servers. There was no bespoke configuration required:



Typical configuration such as Membership Rules and Access Credentials were configured and the agents for Management Center, Application Control, Environment Manager, and Performance Manager deployed. Again, no bespoke configuration was required.

Overall Configuration

The diagram below provides an overview of the configuration of Ivanti User Workspace Manager and Microsoft Azure platform in a load-balanced environment:



Additional Reading

- Choose a cloud SQL Server option: Azure SQL (PaaS) Database or SQL Server on Azure VMs (IaaS)
- Windows Virtual Machines Home