



# XCubeSAN SANOS RESTful API Software Manual

## Applicable Models:

XS5224D, XS5216D, XS5212D, XS5212S, XS5226D, XS5226S  
XS3224D, XS3224S, XS3216D, XS3216S, XS3212D, XS3212S  
XS3226D, XS3226S, XS1224D, XS1224S, XS1216D, XS1216S  
XS1212D, XS1212S, XS1226D, XS1226S



QSAN Technology, Inc.  
[www.QSAN.com](http://www.QSAN.com)



## **Copyright**

© Copyright 2019 QSAN Technology, Inc. All rights reserved. No part of this document may be reproduced or transmitted without written permission from QSAN Technology, Inc.

## **April 2019**

This edition applies to QSAN XCubeSAN SANOS (SAN Operating System) 4.0. QSAN believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

## **Trademarks**

QSAN, the QSAN logo, XCubeSAN, and QSAN.com are trademarks or registered trademarks of QSAN Technology, Inc.

Microsoft, Windows, Windows Server, and Hyper-V are trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Linux is a trademark of Linus Torvalds in the United States and/or other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Mac and OS X are trademarks of Apple Inc., registered in the U.S. and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

VMware, ESXi, and vSphere are registered trademarks or trademarks of VMware, Inc. in the United States and/or other countries.

Citrix and Xen are registered trademarks or trademarks of Citrix Systems, Inc. in the United States and/or other countries.

Other trademarks and trade names used in this document to refer to either the entities claiming the marks and names or their products are the property of their respective owners.

## Notices

---

This XCubeSAN SANOS 4.0 user's manual is applicable to the following XCubeSAN models:

### XCubeSAN Storage System 4U 19" Rack Mount Models

Model Name	Controller Type	Form Factor, Bay Count, and Rack Unit
XS5224D	Dual Controller	LFF 24-disk 4U Chassis
XS3224D	Dual Controller	LFF 24-disk 4U Chassis
XS3224S	Single Controller	LFF 24-disk 4U Chassis
XS1224D	Dual Controller	LFF 24-disk 4U Chassis
XS1224S	Single Controller	LFF 24-disk 4U Chassis

### XCubeSAN Storage System 3U 19" Rack Mount Models

Model Name	Controller Type	Form Factor, Bay Count, and Rack Unit
XS5216D	Dual Controller	LFF 16-disk 3U Chassis
XS3216D	Dual Controller	LFF 16-disk 3U Chassis
XS3216S	Single Controller	LFF 16-disk 3U Chassis
XS1216D	Dual Controller	LFF 16-disk 3U Chassis
XS1216S	Single Controller	LFF 16-disk 3U Chassis

### XCubeSAN Storage System 2U 19" Rack Mount Models

Model Name	Controller Type	Form Factor, Bay Count, and Rack Unit
XS5212D	Dual Controller	LFF 12-disk 2U Chassis
XS5212S	Single Controller	LFF 12-disk 2U Chassis
XS3212D	Dual Controller	LFF 12-disk 2U Chassis
XS3212S	Single Controller	LFF 12-disk 2U Chassis
XS1212D	Dual Controller	LFF 12-disk 2U Chassis
XS1212S	Single Controller	LFF 12-disk 2U Chassis
XS5226D	Dual Controller	SFF 26-disk 2U Chassis
XS5226S	Single Controller	SFF 26-disk 2U Chassis
XS3226D	Dual Controller	SFF 26-disk 2U Chassis
XS3226S	Single Controller	SFF 26-disk 2U Chassis
XS1226D	Dual Controller	SFF 26-disk 2U Chassis

XS1226S	Single Controller	SFF 26-disk 2U Chassis
---------	-------------------	------------------------

Information contained in this manual has been reviewed for accuracy. But it could include typographical errors or technical inaccuracies. Changes are made to the document periodically. These changes will be incorporated in new editions of the publication. QSAN may make improvements or changes in the products. All features, functionality, and product specifications are subject to change without prior notice or obligation. All statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products.

All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

# Table of Contents

---

<b>Notices .....</b>	<b>i</b>
<b>Preface .....</b>	<b>vii</b>
About This Manual .....	vii
Related Documents .....	vii
Technical Support .....	vii
Information, Tip and Caution .....	viii
Conventions .....	viii
<b>1. RESTful API Overview .....</b>	<b>1</b>
1.1. What is RESTful AP .....	1
1.2. How RESTful APIs work .....	1
<b>2. Getting Started with RESTful API .....</b>	<b>2</b>
<b>3. RESTful API Sets .....</b>	<b>3</b>
3.1. Login .....	3
3.2. Help .....	4
3.3. Dashboard .....	5
3.3.1. System Information .....	5
3.3.2. Hardware Monitor .....	6
3.4. List Event Logs .....	7
3.5. Storage .....	8
3.5.1. Disk Information .....	9
3.5.2. Disk S.M.A.R.T. ....	10
3.5.3. Pool Information .....	11
3.5.4. Volume Information .....	12
3.5.5. Snapshot Information .....	14
3.5.6. List Targets .....	15
3.5.7. List Target LUNs .....	15
3.5.8. List LUN Mappings .....	16
3.5.9. Take a Snapshot .....	17
3.5.10. Set Snapshot Space .....	18
3.5.11. Map a LUN .....	19
3.5.12. Unmap the LUN .....	20
<b>4. Support and Other Resources .....</b>	<b>21</b>
4.1. Getting Technical Support .....	21
4.2. Online Customer Support .....	23



- 4.3. Accessing Product Updates..... 27
- 4.4. Documentation Feedback ..... 28
- Appendix..... 29**
- Glossary and Acronym List..... 29
- End-User License Agreement (EULA) ..... 32

## Figures

---

Figure 4-1	Download Service Package in the SANOS UI .....	22
Figure 4-2	Appearance of a Console Cable .....	23
Figure 4-3	Connect the Console Cable .....	23
Figure 4-4	The Procedures of Setup Serial Console by HyperTerminal .....	25
Figure 4-5	The Procedures of Setup Serial Console by PuTTY .....	27

## Tables

---

Table 3-1	Login Parameters .....	3
Table 3-2	List Event Parameters .....	7
Table 3-3	Snapshot Information Parameters.....	14
Table 3-4	List Target LUN Parameters .....	16
Table 3-5	Take Snapshot Parameters .....	18
Table 3-6	Set Snapshot Space Parameters .....	18
Table 3-7	Map LUN Parameters.....	19
Table 3-8	Unmap LUN Parameters .....	20



# Preface

---

## About This Manual

This manual provides technical guidance for designing and implementing QSAN XCubeSAN series SAN system, and it is intended for use by system administrators, SAN designers, storage consultants, or anyone who has purchased these products and is familiar with servers and computer networks, network administration, storage system installation and configuration, storage area network management, and relevant protocols.

## Related Documents

There are related documents which can be downloaded from the website.

- [All XCubeSAN Documents](#)
- [XCubeSAN QIG \(Quick Installation Guide\)](#)
- [XCubeSAN Hardware Manual](#)
- [XCubeSAN Configuration Worksheet](#)
- [XCubeSAN SANOS 4.0 Software Manual](#)
- [Compatibility Matrix](#)
- [White Papers](#)
- [Application Notes](#)

## Technical Support

Do you have any questions or need help trouble-shooting a problem? Please contact QSAN Support, we will reply to you as soon as possible.

- Via the Web: <https://qsan.com/support>
- Via Telephone: +886-2-77206355  
(Service hours: 09:30 - 18:00, Monday - Friday, UTC+8)
- Via Skype Chat, Skype ID: qsan.support  
(Service hours: 09:30 - 02:00, Monday - Friday, UTC+8, Summer time: 09:30 - 01:00)
- Via Email: [support@qsan.com](mailto:support@qsan.com)

## Information, Tip and Caution

This manual uses the following symbols to draw attention to important safety and operational information.



**INFORMATION:**

INFORMATION provides useful knowledge, definition, or terminology for reference.



**TIP:**

TIP provides helpful suggestions for performing tasks more effectively.



**CAUTION:**

CAUTION indicates that failure to take a specified action could result in damage to the system.

## Conventions

The following table describes the typographic conventions used in this manual.

Conventions	Description
<b>Bold</b>	Indicates text on a window, other than the window title, including menus, menu options, buttons, fields, and labels. Example: Click the <b>OK</b> button.
<i>&lt;Italic&gt;</i>	Indicates a variable, which is a placeholder for actual text provided by the user or system. Example: copy <i>&lt;source-file&gt;</i> <i>&lt;target-file&gt;</i> .
[ ] square brackets	Indicates optional values. Example: [ a   b ] indicates that you can choose a, b, or nothing.
{ } braces	Indicates required or expected values. Example: { a   b } indicates that you must choose either a or b.
vertical bar	Indicates that you have a choice between two or more options or

	arguments.
/ Slash	Indicates all options or arguments.
underline	Indicates the default value. Example: [ <u>a</u>   b ]



# 1. RESTful API Overview

---

QSAN meets the trend of the times and supports the RESTful API which is becoming the most popular API design specification.

## 1.1. What is RESTful AP

A RESTful API is based on REST (REpresentational State Transfer) technology, an architectural style and approach to communications often used in web services development. A RESTful API is an API (Application Program Interface) that uses HTTP requests to GET, POST, PUT, and DELETE data.

RESTful API design is designed to take advantage of existing protocols. It usually takes advantage of HTTP when used for Web APIs. This means that developers do not need to install libraries or additional software in order to take advantage of a REST API design. REST API is notable for its incredible layer of flexibility and leverages less bandwidth, making it more suitable for internet usage. Since data is not tied to methods and resources, REST has the ability to handle multiple types of calls, return different data formats and even change structurally with the correct implementation of hypermedia.

## 1.2. How RESTful APIs work

A RESTful API breaks down a transaction to create a series of small modules. Each module addresses a particular underlying part of the transaction. This modularity provides developers with a lot of flexibility, but it can be challenging for developers to design from scratch.

A RESTful API explicitly takes advantage of HTTP methodologies. They use GET to retrieve a resource; PUT to change the state of or update a resource, which can be an object, file or block; POST to create that resource; and DELETE to remove it.

## 2. Getting Started with RESTful API

---

There are some useful API tools to test before developing an application using the RESTful API. API testing is a software testing type which focuses on the determination if the developed APIs meet expectations regarding the functionality, reliability, performance, and security of the application. The following provides some API testing tools that testing teams can select to suit your needs.

### **Postman**

Website: <https://www.getpostman.com/>

Postman is an easy-to-use REST client and rich interface which makes it easy to use.

### **SoapUI**

Website: <https://www.soapui.org/>

SoapUI is a headless functional testing tool dedicated to API testing, allowing users to test REST and SOAP APIs and Web Services easily.

### **Katalon Studio**

Website: <https://www.katalon.com>

Katalon Studio is a free test automation tool for API, Web and Mobile applications.

## 3. RESTful API Sets

---

This chapter is to help you find an API by name. Each API topic includes one or more of the following sections:

<b>API</b>	The Application Program Interface
<b>Description</b>	The API's purpose and note about usage
<b>Parameters</b>	Descriptions of API's parameters
<b>Example</b>	One or more examples of API's usage

The usage of each API is described in the following.

### 3.1. Login

#### API

POST <baseurl>/rest/login/account/\$username/password/\$password

#### Description

Login the system.

#### Parameters

Table 3-1 Login Parameters

Name	Type	Description	Default
\$username	string	Login account	
\$password	string	Login password	

#### Example

First, you have to login the system, then you can execute the RESTful APIs.

```
POST http://<IP or FQDN>/rest/login/account/admin/password/1234

{
  "success": true,
  "msg": "",
  "data": []
}
```

## 3.2. Help

### API

GET <baseurl>/rest/

### Description

Show all APIs.

### Parameters

None.

### Example

Display all RESTfull APIs of the system.

```
GET http://<IP or FQDN>/rest

{
  "success": false,
  "msg": "Invalid param",
  "data": [
    "=====GET=====",
    "/rest/dashboard/systeminfo",
    "/rest/dashboard/hwmonitor",
    "/rest/log/list/$page_num",
    "/rest/storage/diskinfo",
    "/rest/storage/disksmart",
    "/rest/storage/poolinfo",
    "/rest/storage/volumeinfo",
    "/rest/storage/snapshot/vdid/$vdid",
    "/rest/storage/availltarget",
    "/rest/storage/availlun/targetid/$target_id",
    "/rest/storage/lunmappinginfo",
    "=====GET=====",
    "",
    "",
    "=====",
    "/rest/login/account/$username/password/$password",
    "/rest/storage/snapshot/vdid/$vdid/name/$snapshotname",
    "/rest/storage/snapspace/vdid/$vdid/sizegb/$sizegb",
  ]
}
```



```
"/rest/storage/maplun/vdid/$vdid/targetid/$targetid/lunid/$lunid/host/$host",
    "/rest/storage/unmaplun/targetid/$targetid/lunid/$lunid/host/$host",
    "====POST===="
  ]
}
```

## 3.3. Dashboard

This section includes the following API sets.

### GET

- **sysname**: Show the system information.
- **hwmonitor**: Show the hardware monitoring information.

### 3.3.1. System Information

#### API

GET <baseurl>/rest/dashboard/systeminfo

#### Description

Show the system information.

#### Parameters

None.

#### Example

Display the basic information of the system.

```
GET http://<IP or FQDN>/rest/dashboard/systeminfo

{
  "success": true,
  "msg": "",
  "data": {
    "system_name": "XCubeSAN",
    "model_name": XS5216,
    "bpl_id": "QW316",
    "sys_sn": "001378xxxxxx",
    "firmware_version": "XS5216 1.4.0 (build 201904xxxxxx) ",
  }
}
```

```
    "sys_ctrl_status": "Normal",  
    "sys_health": "Good"  
  }  
}
```

## 3.3.2. Hardware Monitor

### API

GET <baseurl>/rest/dashboard/hwmonitor

### Description

Show the hardware monitoring information.

### Parameters

None.

### Example

Show summary status of all enclosures. There are current voltage, temperature, status of fan module, and power supply.

```
GET http://<IP or FQDN>/rest/dashboard/hwmonitor
```

```
{  
  "success": true,  
  "msg": "",  
  "data": {  
    "Enc0": {  
      "C2F": [  
        {  
          "item": "power_module",  
          "value": "Good"  
        },  
        {  
          "item": "flash_module",  
          "value": "Good"  
        }  
      ],  
      "hardware": [  
        {  
          "item": "power_supply",  
          "value": "Good"  
        },  
        {  
          "item": "fan_module",  
          "value": "Good"  
        }  
      ]  
    }  
  }  
}
```

```

    ],
    "CTR1": [
      {
        "item": "Voltage",
        "value": "Good"
      },
      {
        "item": "Temperature",
        "value": "Good"
      }
    ],
    "BPL": [
      {
        "item": "Voltage",
        "value": "Good"
      },
      {
        "item": "Temperature",
        "value": "Good"
      }
    ],
    "CTR2": [
      {
        "item": "Voltage",
        "value": "Good"
      },
      {
        "item": "Temperature",
        "value": "Good"
      }
    ]
  ]
}

```

### 3.4. List Event Logs

#### API

GET <baseurl>/rest/log/list/\$page\_num

#### Description

Show the system event logs.

#### Parameters

Table 3-2 List Event Parameters

Name	Type	Description	Default
\$page_num	integer	Even log page number	

## Example

Display the event logs for the first page.

```
GET http://<IP or FQDN>/rest/log/list/1

{
  "success": true,
  "msg": "",
  "data": [
    {
      "key": 0,
      "type": "INFO:",
      "time": "Wed, Mar 27, 2019 15:34:24",
      "content": "[CTR1] admin login from 192.168.100.100 via web UI."
    },
    {
      "key": 1,
      "type": "INFO:",
      "time": "Wed, Mar 27, 2019 15:09:54",
      "content": "[CTR1] The pool 'Go2' has been created."
    },
    . . . (continue)
    {
      "key": 19,
      "type": "INFO:",
      "time": "Wed, Mar 27, 2019 10:08:17",
      "content": "[CTR1] The pool 'p1' has been deleted."
    }
  ]
}
```

## 3.5. Storage

This section includes the following API sets.

### GET

- **diskinfo**: Show the system information.
- **disksmart**: Show the hardware monitoring information.
- **poolinfo**: Show the system information.
- **volumeinfo**: Show the hardware monitoring information.
- **snapshot**: Show the system information.
- **availtarget**: Show the hardware monitoring information.
- **avallun**: Show the system information.

- **lunmappinginfo**: Show the hardware monitoring information.

## POST

- **snapshot**: Show the system information.
- **snapspace**: Show the hardware monitoring information.
- **maplun**: Show the system information.
- **unmaplun**: Show the hardware monitoring information.

### 3.5.1. Disk Information

#### API

GET <baseurl>/rest/storage/diskinfo

#### Description

Show the disk information.

#### Parameters

None.

#### Example

Display all disk information of all enclosures.

```
GET http://<IP or FQDN>/rest/storage/diskinfo
```

```
{
  "success": true,
  "msg": "",
  "data": [
    {
      "name": "pd",
      "item": {
        "ID": 136855096,
        "No.": 1,
        "Enc ID": 0,
        "Enc No.": 1,
        "Slot": 11,
        "WWN": "5000cca04d2dd51f",
        "Size(MB)": 190526,
        "Size(GB)": 186,
        "RG ID": 1073512122,
        "RG name": "Go2",
        "SSD": "Yes",
        "SED": "ISE",
```

```

        "SED Status": "",
        "ISE Status": "Normal",
        "SED Hint": "",
        "RAID Set": 1,
        "SpinRate": 1,
        "Sector": 512,
        "Status": "Online",
        "Health": "Good",
        "Error alert": "No",
        "Read errors": "No",
        "Usage": "RD",
        "Vendor": "HGST",
        "Serial": "0PVU6D6A",
        "Model": "HUSMM1620ASS200",
        "FW ver.": "A2C0",
        "Rate": "SAS SSD 12.0Gb/s",
        "Write cache": "Enabled",
        "MAID 2.0": "N/A",
        "Park head": "N/A",
        "Reduce speed": "N/A",
        "Standby": "Disabled",
        "Readahead": "Enabled",
        "CmdQue": "Enabled"
    },
    . . . (continue)
]
}

```

### 3.5.2. Disk S.M.A.R.T.

#### API

GET <baseurl>/rest/storage/disksmart

#### Description

Show the disk S.M.A.R.T. information.

#### Parameters

None.

#### Example

Display disk S.M.A.R.T. information of all enclosures.

```

{
  "success": true,

```

```

"msg": "",
"data": {
  "136855096": {
    "write_err_corr_wi_possible_delays": 0,
    "total_write_err": 0,
    "total_write_err_corr": 0,
    "total_write_err_corr_by_algo": 0,
    "total_bytes_written": 603695221807104,
    "total_uncorr_write_err": 0,
    "read_err_corr_by_ecc_hw_method": 0,
    "read_err_corr_wi_possible_delays": 0,
    "total_read_err": 0,
    "total_read_err_corr": 0,
    "total_read_err_corr_by_algo": 0,
    "total_bytes_read": 332840023195136,
    "total_uncorr_read_err": 0,
    "non_medium_err": 0,
    "temperature": "26C / 78.8F</value>",
    "ref_temperature": "70C / 158F</value>",
    "specified_cycle_cnt_over_dev_lifetime": 0,
    "accumulated_start_stop_cycles": 0,
    "accumulated_load_unload_cnt_over_dev_lifetime": 0,
    "accumulated_load_unload_cycles": 0,
    "blocks_sent_to_initiator": 277543782944079872,
    "blocks_received_from_initiator": 0,
    "blocks_read_from_cache_and_send_to_initiator": 0,
    "IO_cmd_wi_size_smaller_than_segment_size": 0,
    "IO_cmd_wi_size_larger_than_segment_size": 0,
    "remain_life": 90
  },
  . . . (continue)
}
}

```

### 3.5.3. Pool Information

#### API

GET <baseurl>/rest/storage/poolinfo

#### Description

Show the pool information.

#### Parameters

None.

#### Example

Display all pools of the system.

```
GET http://<IP or FQDN>/rest/storage/poolinfo
```

```
{
  "success": true,
  "msg": "",
  "data": [
    {
      "name": "rg",
      "item": {
        "ID": 1057193661,
        "No.": 1,
        "Name": "Pool-1",
        "Total(MB)": 2288770,
        "Total(GB)": 2235,
        "Free(MB)": 2186370,
        "Free(GB)": 2135,
        "Avail(MB)": 2186370,
        "Avail(GB)": 2135,
        "Secure": "Disabled",
        "Thin": "Disabled",
        "Tier": "Disabled",
        "#PD": 3,
        "#VD": 1,
        "Status": "Online",
        "Health": "Good",
        "Attributes": 2,
        "RC": 1,
        "RAID": "RAID 5",
        "Owner": "CTR 1",
        "POwner": "CTR 1",
        "HA mode": "A-A",
        "Enc No.": 0,
        "Enclosure": "Local",
        "SCRUB_SUPPORT": 1,
        "PD": "13 14 16 ",
        "D-SP": "",
        "Mem_requiredP": 4
      }
    }
  ]
}
```

### 3.5.4. Volume Information

#### API

```
GET <baseurl>/rest/storage/volumeinfo
```

#### Description

Show the system information.



## Parameters

None.

## Example

Display all volumes of the system.

```
GET http://<IP or FQDN>/rest/storage/volumeinfo
```

```
{
  "success": true,
  "msg": "",
  "data": [
    {
      "name": "udv",
      "item": {
        "ID": 1059612626,
        "No.": 1,
        "Name": "Vol-1",
        "WWN": "20330013780a23c0",
        "Size(MB)": 102400,
        "Size(GB)": 100,
        "Target ID": 4294967295,
        "Clone": "N/A",
        "Write": "WB",
        "Priority": "MD",
        "Bg rate": 4,
        "Readahead": "Disabled",
        "AV-media": "Disabled",
        "Reclamation": "Disabled",
        "Fast rebuild": "Disabled",
        "Tier Policy": 0,
        "Prov. size(MB)": 102400,
        "Status": "Initiating",
        "Health": "Optimal",
        " %": 25,
        "Strip(KB)": 64,
        "RAID": "RAID 5",
        "#LUN": 0,
        "Snap space(MB)": "0/0",
        "#Snap": 0,
        "Type": "RAID",
        "RG ID": 1057193661,
        "RG name": "Pool-1",
        "CV ID": 1,
        "CV (MB)": 6645,
        "Scrub support": 1,
        "Block size": 512,
        "Schedule": ""
      }
    }
  ]
}
```

## 3.5.5. Snapshot Information

### API

GET <baseurl>/rest/storage/snapshot/vdid/\$vdid

### Description

Show the snapshots of the volume.

### Parameters

Table 3-3 Snapshot Information Parameters

Name	Type	Description
\$vdid	integer	Volume ID

### Example

Display all taken snapshots of the volume.

```
GET http://<IP or FQDN>/rest/storage/snapshot/vdid/1059612626
```

```
{
  "success": true,
  "msg": "",
  "data": [
    {
      "name": "udv_snap",
      "item": {
        "Tag": 0,
        "No.": 1,
        "Name": "Snap-1",
        "Used(MB)": 0,
        "Status": "N/A",
        "Health": "Good",
        "Exposure": "No",
        "VD ID": 4294967295,
        "Right": "N/A",
        "#LUN": "N/A",
        "Created time": "Fri Mar 29 16:45:05 2019",
        "Deleted time": "Thu Jan 1 08:00:00 1970"
      }
    }
  ]
}
```

### 3.5.6. List Targets

#### API

GET <baseurl>/rest/storage/availtarget

#### Description

List all available targets.

#### Parameters

None.

#### Example

Display all targets of the system.

```
GET http://<IP or FQDN>/rest/storage/availtarget

{
  "success": true,
  "msg": "",
  "data": [
    {
      "target_name": "iqn.2004-08.com.qsan:xs5216-000d42340:dev0.ctr1",
      "target_id": 0
    },
    {
      "target_name": "iqn.2004-08.com.qsan:xs5216-000d42340:dev1.ctr1",
      "target_id": 1
    },
    . . . (continue)
    {
      "target_name": "iqn.2004-08.com.qsan:xs5216-000d42340:dev127.ctr1",
      "target_id": 127
    }
  ]
}
```

### 3.5.7. List Target LUNs

#### API

GET <baseurl>/storage/availlun/targetid/\$target\_id

## Description

List all available LUNs of the target.

## Parameters

Table 3-4 List Target LUN Parameters

Name	Type	Description
\$target_id	integer	Target ID

## Example

Display all LUNs of the target.

```
GET http://<IP or FQDN>/rest/storage/availlun/targetid/0

{
  "success": true,
  "msg": "",
  "data": [
    {
      "lun_name": "LUN 0",
      "lun_id": 0
    },
    {
      "lun_name": "LUN 1",
      "lun_id": 1
    },
    . . . (continue)
    {
      "lun_name": "LUN 254",
      "lun_id": 254
    }
  ]
}
```

### 3.5.8. List LUN Mappings

#### API

GET <baseurl>/rest/storage/lunmappinginfo

#### Description

Show all LUN mappings.

### Parameters

None.

### Example

Display all LUN mappings.

```
GET http://<IP or FQDN>/rest/storage/lunmappinginfo

{
  "success": true,
  "msg": "",
  "data": [
    {
      "name": "ace",
      "item": {
        "Host": "*",
        "Target": 0,
        "LUN": 0,
        "VD ID": 1059612626,
        "Permission": "Read write",
        "VD name": "Vol-1",
        "#Session": 0
      }
    }
  ]
}
```

## 3.5.9. Take a Snapshot

### API

POST <baseurl>/rest/storage/snapshot/vdid/\$vdid/name/\$snapshotname

### Description

Take a snapshot for the volume.

### Parameters

Table 3-5 Take Snapshot Parameters

Name	Type	Description
\$vdid	integer	Volume ID
\$snapshotname	string	Snapshot name

### Example

Take a snapshot named “Snap-1” for the volume ID “1059612626”.

```
POST http://<IP or FQDN>/rest/snapshot/vdid/1059612626/name/Snap-1
{
  "success": true,
  "msg": "",
  "data": []
}
```

## 3.5.10. Set Snapshot Space

### API

POST <baseurl>/rest/storage/snapspace/vdid/\$vdid/sizegb/\$sizegb

### Description

Set snapshot space 20GB for the volume.

### Parameters

Table 3-6 Set Snapshot Space Parameters

Name	Type	Description
\$vdid	integer	Volume ID
\$sizegb	integer	Snapshot space size in GB

### Example

Set snapshot space for the volume ID “1059612626”.

```
POST http://<IP or FQDN>/rest/storage/snapspace/vdid/1059612626/sizegb/20

{
  "success": true,
  "msg": "",
  "data": []
}
```

### 3.5.11. Map a LUN

#### API

POST

<baseurl>/rest/storage/maplun/vdid/\$vdid/targetid/\$targetid/lunid/\$lunid/host/\$host

#### Description

Map a LUN for the volume.

#### Parameters

Table 3-7 Map LUN Parameters

Name	Type	Description
\$vdid	integer	Volume ID
\$target_id	integer	Target ID
\$lun_id	integer	LUN ID
\$host	string	Allowed hosts

#### Example

Map a LUN for the volume ID "1059612626" to target ID "0", LUN ID "0", and allowed hosts "\*".

```
POST http://<IP or FQDN>
/rest/storage/maplun/vdid/1059612626/targetid/0/lunid/0/host/*

{
  "success": true,
  "msg": "",
  "data": []
}
```

## 3.5.12. Unmap the LUN

### API

POST <baseurl>/rest/storage/unmaplun/targetid/\$targetid/lunid/\$lunid/host/\$host"

### Description

Unmap the LUN.

### Parameters

Table 3-8 Unmap LUN Parameters

Name	Type	Description
\$target_id	integer	Target ID
\$lun_id	integer	LUN ID
\$host	string	Allowed hosts

### Example

Unmap the LUN with target ID "0", LUN ID "0", and allowed hosts "\*".

```
POST http://<IP or FQDN> /rest/storage/unmaplun/targetid/0/lunid/0/host/*

{
  "success": true,
  "msg": "",
  "data": []
}
```



## 4. Support and Other Resources

---

### 4.1. Getting Technical Support

After installing your device, locate the serial number on the sticker located on the side of the chassis or from the SANOS UI -> **MAINTINANCE** > **System Information** and use it to register your product at <https://partner.qsan.com/> (End-User Registration). We recommend registering your product in QSAN partner website for firmware updates, document download, and latest news in eDM. To contact QSAN Support, please use the following information.

- Via the Web: <https://qsan.com/support>
- Via Telephone: +886-2-77206355  
(Service hours: 09:30 - 18:00, Monday - Friday, UTC+8)
- Via Skype Chat, Skype ID: qsan.support  
(Service hours: 09:30 - 02:00, Monday - Friday, UTC+8, Summer time: 09:30 - 01:00)
- Via Email: [support@qsan.com](mailto:support@qsan.com)

#### Information to Collect

- Product name, model or version, and serial number
- Operating system name and version
- Firmware version
- Error messages or capture screenshots
- Product-specific reports and logs
- Add-on products or components installed
- Third-party products or components installed

#### Information for Technical Support

The following system information is necessary for technical support. Please refer to following for what and where to get the information of your XCubeSAN Series model.

If the technical support requests you to download the Service Package, please navigate in the SANOS UI -> **SYSTEM SETTING** -> **Maintenance** -> **System Information**, and then click the **Download Service Package** button to download. Then the system will automatically generate a zip file the default download location of your web browser.

System	
Item	Information
System Name	XS5216-D40000
Model Name	XS5216
System Serial Number	QW3160160I0021001
System Controller Status	Dual Controller, Active-Active
Master Controller	Controller 1
Controller 1	
Item	Information
Controller Serial Number	50013780080A0440
CPU	Intel(R) Xeon(R) Processor D-1500 4 Cores
Memory	8 GB
Host Card Slot 1	Empty
Host Card Slot 2	Empty
Firmware Version	1.2.0
SAS IOC Firmware Version	07.00.01.00
SAS Expander Firmware Version	1000
Controller 2	
Item	Information
Controller Serial Number	50013780080A0380
CPU	Intel(R) Xeon(R) Processor D-1500 4 Cores
Memory	8 GB
Host Card Slot 1	Empty
Host Card Slot 2	Empty
Firmware Version	1.2.0
SAS IOC Firmware Version	07.00.01.00
SAS Expander Firmware Version	1000
Backplane	
Item	Information
Backplane Serial Number	001378D40000
Backplane ID	QW316
MCU Version	1.2.0
Licenses	
Item	Information
SSD Cache	Enabled
Auto Tiering	Enabled
<a href="#">Download Service Package</a>	

Figure 4-1 Download Service Package in the SANOS UI

## 4.2. Online Customer Support

For better customer support, every XCubeSAN series models include the console cable (two for dual controller models), one for single controller models) for online support. Please follow the procedures below to setup the online help environment for QSAN support team.

The following procedure will help you to setup the serial console via the console cable that is enclosed in the shipping carton. The following image is the appearance of the console cable.



Figure 4-2 Appearance of a Console Cable

### Procedures to Setup the Serial Console

1. Setup the serial cable between the controller and one server/host like in the below image.

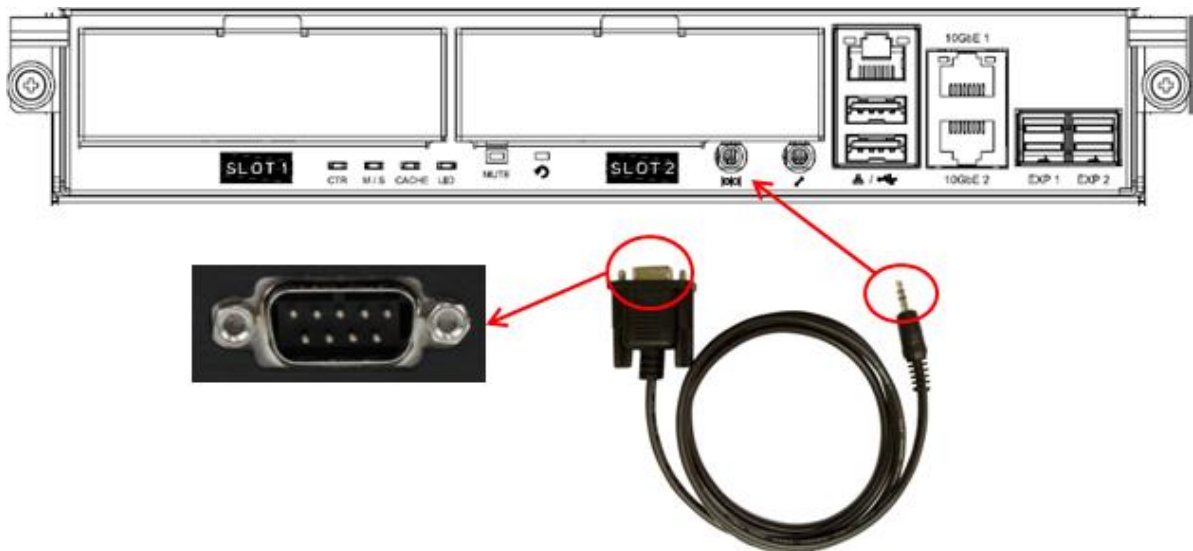


Figure 4-3 Connect the Console Cable

2. You must use terminal software such as HyperTerminal or Putty to open the console after the connection is made.

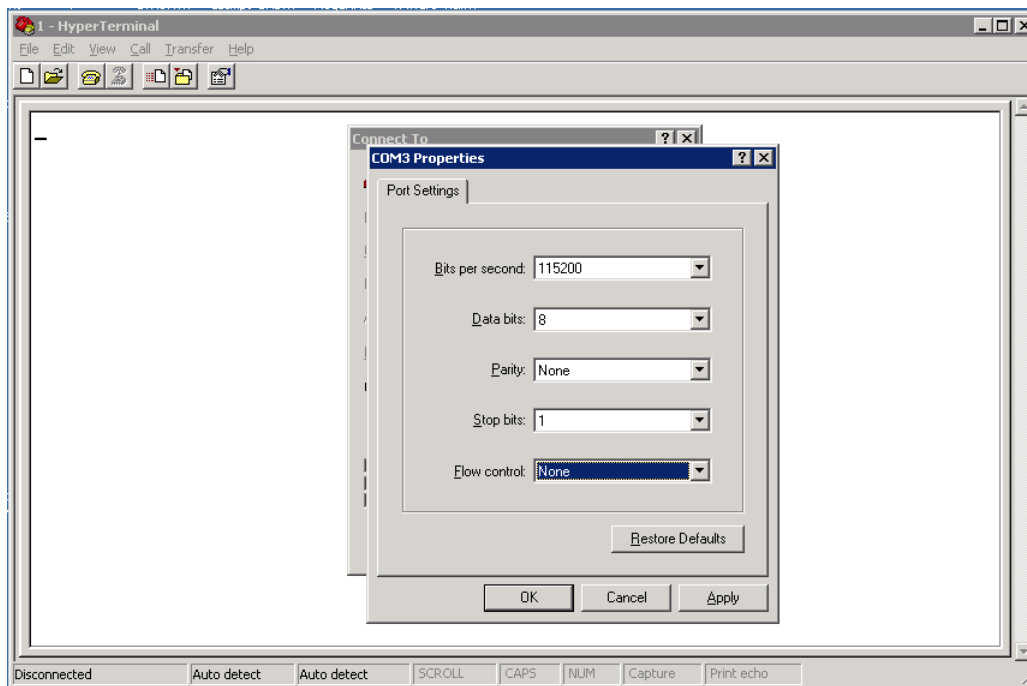
**INFORMATION:**

For more information about terminal software, please refer to

HyperTerminal: <http://www.hilgraeve.com/hyperterminal/>

PuTTY: <http://www.putty.org/>

3. Here we first demonstrate HyperTerminal. The console settings are on the following.  
Baud rate: 115200, 8 data bit, no parity, 1 stop bit, and no flow control  
Terminal type: vt100



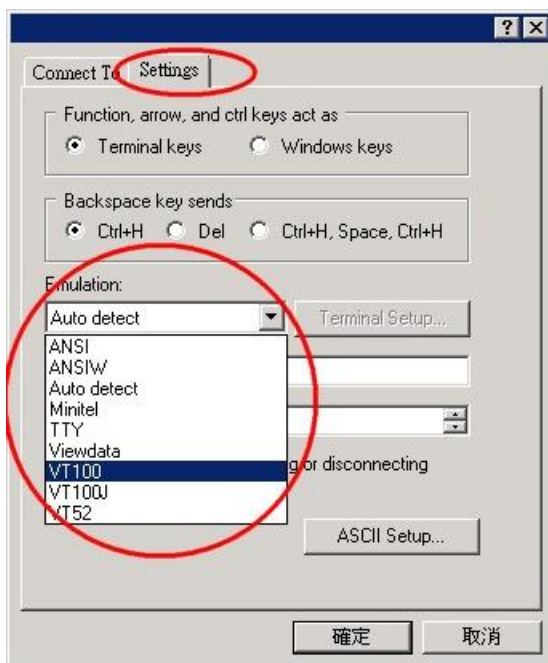
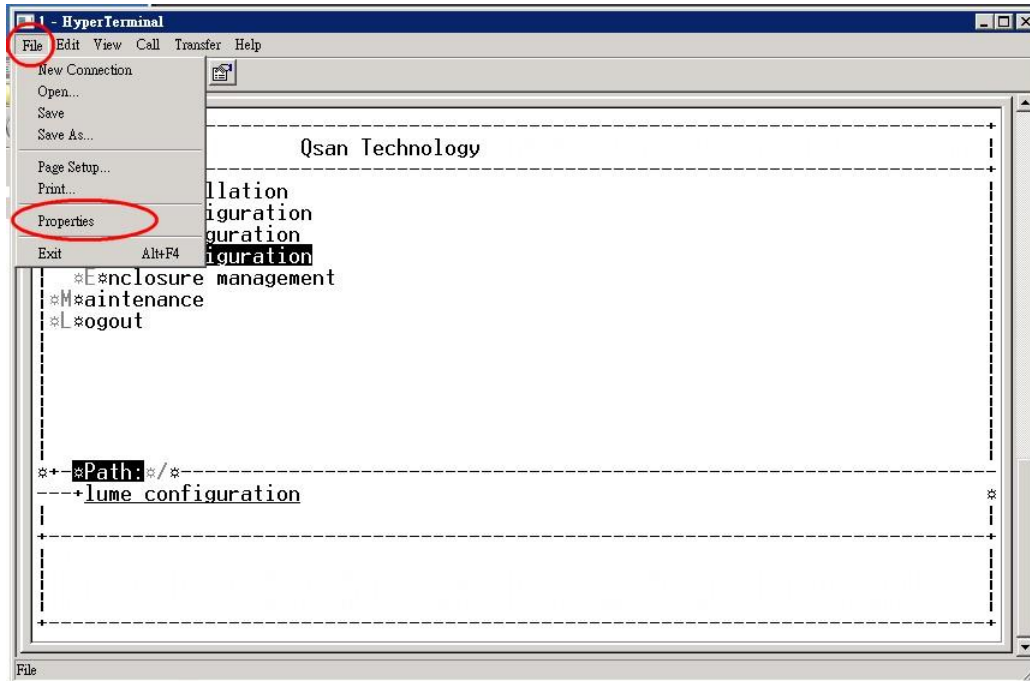
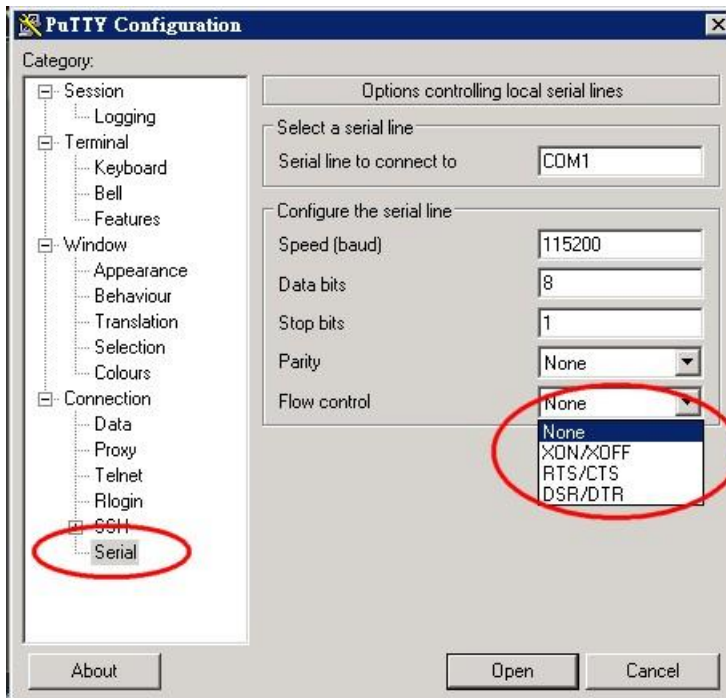
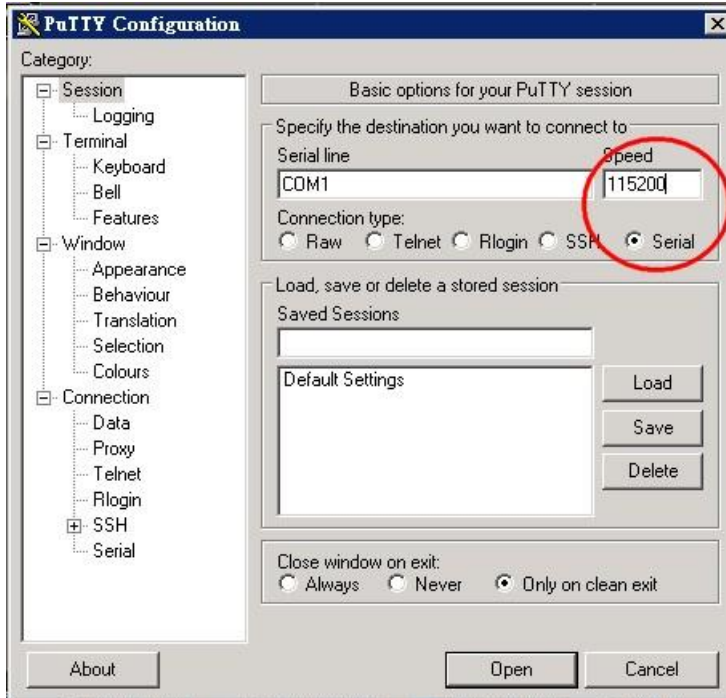


Figure 4-4 The Procedures of Setup Serial Console by HyperTerminal

4. If you are using PuTTY instead, please refer to below



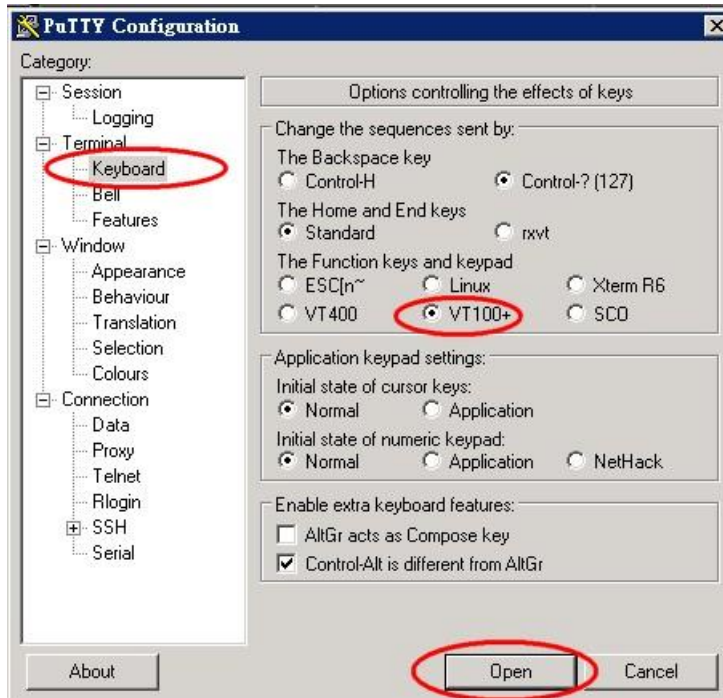


Figure 4-5 The Procedures of Setup Serial Console by PuTTY

5. Users should be able to login the controller system via console cable by following the procedures above.

### Setup the Connection for Online Support

Following is the procedure to setup the connection for online support via TeamViewer:

1. Please download the TeamViewer from following hyper link:  
<https://www.teamviewer.com/en/download/>
2. Install TeamViewer.
3. Please provide the ID/password showed on the application to QSAN support team member to join the online support session.

## 4.3. Accessing Product Updates

To download product updates, please visit QSAN website:

<https://qsan.com/download>

## 4.4. Documentation Feedback

QSAN is committed to providing documentation that meets and exceeds your expectations. To help us improve the documentation, email any errors, suggestions, or comments to [docsfeedback@qsan.com](mailto:docsfeedback@qsan.com).

When submitting your feedback, include the document title, part number, revision, and publication date located on the front cover of the document.



# Appendix

---

## Glossary and Acronym List

### Common Terminology

Item	Description
RAID	Redundant Array of Independent Disks. There are different RAID levels with different degree of data protection, data availability, and performance to host environment.
Disk	The Physical Disk belongs to the member disk of one specific RAID group.
Pool	A collection of removable media. One pool consists of a set of volumes and owns one RAID level attribute.
Volume	Each pool could be divided into several volumes. The volumes from one pool have the same RAID level, but may have different volume capacity.
LUN	Logical Unit Number. A logical unit number (LUN) is a unique identifier which enables it to differentiate among separate devices (each one is a logical unit).
WebUI	Web User Interface.
WT	Write-Through cache-write policy. A cache technique in which the completion of a write request is not signaled until data is safely stored in non-volatile media. Each data is synchronized in both data cache and accessed physical disks.
WB	Write-Back cache-write policy. A cache technique in which the completion of a write request is signaled as soon as the data is in cache and actual writing to non-volatile media occurs at a later time. It speeds up system write performance but needs to bear the risk where data may be inconsistent between data cache and the physical disks in one short time interval.
RO	Set the volume to be Read-Only.
DS	Dedicated Spare disks. The spare disks are only used by one specific

	RAID group. Others could not use these dedicated spare disks for any rebuilding purpose.
LS	Local Spare disks. The spare disks are only used by the RAID groups of the local enclosure. Other enclosure could not use these local spare disks for any rebuilding purpose.
GS	Global Spare disks. It is shared for rebuilding purpose. If some RAID groups need to use the global spare disks for rebuilding, they could get the spare disks out from the common spare disks pool for such requirement.
DG	DeGraded mode. Not all of the array's member disks are functioning, but the array is able to respond to application read and write requests to its virtual disks.
SCSI	Small Computer System Interface
SAS	Serial Attached SCSI
S.M.A.R.T.	Self-Monitoring Analysis and Reporting Technology
WWN	World Wide Name
HBA	Host Bus Adapter
SES	SCSI Enclosure Services
NIC	Network Interface Card
BBM	Battery Backup Module
SCM	Super Capacitor Module

## FC / iSCSI / SAS Terminology

Item	Description
FC	Fibre Channel
FC-P2P	Point-to-Point
FC-AL	Arbitrated Loop
FC-SW	Switched Fabric
iSCSI	Internet Small Computer Systems Interface
LACP	Link Aggregation Control Protocol
MPIO	Multipath Input/Output
MC/S	Multiple Connections per Session
MTU	Maximum Transmission Unit
CHAP	Challenge Handshake Authentication Protocol. An optional security

	mechanism to control access to an iSCSI storage system over the iSCSI data ports.
iSNS	Internet Storage Name Service
SAS	Serial Attached SCSI

### Dual Controller Terminology

Item	Description
SBB	Storage Bridge Bay. The objective of the Storage Bridge Bay Working Group (SBB) is to create a specification that defines mechanical, electrical and low-level enclosure management requirements for an enclosure controller slot that will support a variety of storage controllers from a variety of independent hardware vendors (“IHVs”) and system vendors.
6G MUX	Bridge board is for SATA II disk to support dual controller mode.



## End-User License Agreement (EULA)

Please read this document carefully before you use our product or open the package containing our product.

YOU AGREE TO ACCEPT TERMS OF THIS EULA BY USING OUR PRODUCT, OPENING THE PACKAGE CONTAINING OUR PRODUCT OR INSTALLING THE SOFTWARE INTO OUR PRODUCT. IF YOU DO NOT AGREE TO TERMS OF THIS EULA, YOU MAY RETURN THE PRODUCT TO THE RESELLER WHERE YOU PURCHASED IT FOR A REFUND IN ACCORDANCE WITH THE RESELLER'S APPLICABLE RETURN POLICY.

### **General**

QSAN Technology, Inc. ("QSAN") is willing to grant you ("User") a license of software, firmware and/or other product sold, manufactured or offered by QSAN ("the Product") pursuant to this EULA.

### **License Grant**

QSAN grants to User a personal, non-exclusive, non-transferable, non-distributable, non-assignable, non-sub-licensable license to install and use the Product pursuant to the terms of this EULA. Any right beyond this EULA will not be granted.

### **Intellectual Property Right**

Intellectual property rights relative to the Product are the property of QSAN or its licensor(s). User will not acquire any intellectual property by this EULA.

### **License Limitations**

User may not, and may not authorize or permit any third party to: (a) use the Product for any purpose other than in connection with the Product or in a manner inconsistent with the design or documentations of the Product; (b) license, distribute, lease, rent, lend, transfer, assign or otherwise dispose of the Product or use the Product in any commercial hosted or service bureau environment; (c) reverse engineer, decompile, disassemble or attempt to discover the source code for or any trade secrets related to the Product, except and only to the extent that such activity is expressly permitted by applicable law notwithstanding this limitation; (d) adapt, modify, alter, translate or create any derivative works of the Licensed Software; (e) remove, alter or obscure any copyright notice or other proprietary rights notice on the Product; or (f) circumvent or attempt to circumvent any methods employed by QSAN to control access to the components, features or functions of the Product.

### **Disclaimer**

QSAN DISCLAIMS ALL WARRANTIES OF PRODUCT, INCLUDING BUT NOT LIMITED TO ANY MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, WORKMANLIKE EFFORT, TITLE, AND NON-INFRINGEMENT. ALL PRODUCTS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. QSAN MAKES NO WARRANTY THAT THE PRODUCT WILL BE FREE OF BUGS, ERRORS, VIRUSES OR OTHER DEFECTS.

IN NO EVENT WILL QSAN BE LIABLE FOR THE COST OF COVER OR FOR ANY DIRECT, INDIRECT, SPECIAL, PUNITIVE, INCIDENTAL, CONSEQUENTIAL OR SIMILAR DAMAGES OR LIABILITIES WHATSOEVER (INCLUDING, BUT NOT LIMITED TO LOSS OF DATA, INFORMATION, REVENUE, PROFIT OR BUSINESS) ARISING OUT OF OR RELATING TO THE USE OR INABILITY TO USE THE PRODUCT OR OTHERWISE UNDER OR IN CONNECTION WITH THIS EULA OR THE PRODUCT, WHETHER BASED ON CONTRACT, TORT (INCLUDING NEGLIGENCE), STRICT LIABILITY OR OTHER THEORY EVEN IF QSAN HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

### **Limitation of Liability**

IN ANY CASE, QSAN'S LIABILITY ARISING OUT OF OR IN CONNECTION WITH THIS EULA OR THE PRODUCT WILL BE LIMITED TO THE TOTAL AMOUNT ACTUALLY AND ORIGINALLY PAID BY CUSTOMER FOR THE PRODUCT. The foregoing Disclaimer and Limitation of Liability will apply to the maximum extent permitted by applicable law. Some jurisdictions do not allow the exclusion or limitation of incidental or consequential damages, so the exclusions and limitations set forth above may not apply.

### **Termination**

If User breaches any of its obligations under this EULA, QSAN may terminate this EULA and take remedies available to QSAN immediately.

### **Miscellaneous**

- QSAN reserves the right to modify this EULA.
- QSAN reserves the right to renew the software or firmware anytime.
- QSAN may assign its rights and obligations under this EULA to any third party without condition.
- This EULA will be binding upon and will inure to User's successors and permitted assigns.

- This EULA shall be governed by and constructed according to the laws of R.O.C. Any disputes arising from or in connection with this EULA, User agree to submit to the jurisdiction of Taiwan Shilin district court as first instance trial.