

# Novell Developer Kit

[www.novell.com](http://www.novell.com)

October 11, 2006

XATTR EXTENSION FOR NOVELL  
STORAGE SERVICES™

# N

**Novell®**

## Legal Notices

Novell, Inc. makes no representations or warranties with respect to the contents or use of this documentation, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to revise this publication and to make changes to its content, at any time, without obligation to notify any person or entity of such revisions or changes.

Further, Novell, Inc. makes no representations or warranties with respect to any software, and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose. Further, Novell, Inc. reserves the right to make changes to any and all parts of Novell software, at any time, without any obligation to notify any person or entity of such changes.

Any products or technical information provided under this Agreement may be subject to U.S. export controls and the trade laws of other countries. You agree to comply with all export control regulations and to obtain any required licenses or classification to export, re-export, or import deliverables. You agree not to export or re-export to entities on the current U.S. export exclusion lists or to any embargoed or terrorist countries as specified in the U.S. export laws. You agree to not use deliverables for prohibited nuclear, missile, or chemical biological weaponry end uses. Please refer to <http://www.novell.com/info/exports/> (<http://www.novell.com/info/exports/>) for more information on exporting Novell software. Novell assumes no responsibility for your failure to obtain any necessary export approvals.

Copyright © 2006 Novell, Inc. All rights reserved. No part of this publication may be reproduced, photocopied, stored on a retrieval system, or transmitted without the express written consent of the publisher.

Novell, Inc. has intellectual property rights relating to technology embodied in the product that is described in this document. In particular, and without limitation, these intellectual property rights may include one or more of the U.S. patents listed at <http://www.novell.com/company/legal/patents/> (<http://www.novell.com/company/legal/patents/>) and one or more additional patents or pending patent applications in the U.S. and in other countries.

Novell, Inc.  
404 Wyman Street, Suite 500  
Waltham, MA 02451  
U.S.A.  
[www.novell.com](http://www.novell.com)

*Online Documentation:* To access the online documentation for this and other Novell developer products, and to get updates, see [developer.novell.com/ndk](http://developer.novell.com/ndk). To access online documentation for Novell products, see [www.novell.com/documentation](http://www.novell.com/documentation).

## **Novell Trademarks**

For Novell trademarks, see the [Novell Trademark and Service Mark list \(http://www.novell.com/company/legal/trademarks/tmlist.html\)](http://www.novell.com/company/legal/trademarks/tmlist.html).

## **Third-Party Materials**

All third-party trademarks are the property of their respective owners.



# Contents

<b>About This Guide</b>	<b>7</b>
<b>1 Concepts</b>	<b>9</b>
1.1 Overview .....	9
1.2 Data Sizes .....	9
1.3 CtimeIsMetadataModTime Command Line Option .....	10
<b>2 Functions</b>	<b>11</b>
getxattr .....	12
listxattr .....	13
removexattr .....	14
setxattr .....	15
<b>3 Structures</b>	<b>17</b>
zNW_metadata_s .....	18
zNW_trustee_s .....	20
<b>4 Values</b>	<b>21</b>
4.1 File Attribute Values .....	21
4.2 Modify Mask Values .....	22
<b>A Revision History</b>	<b>25</b>



# About This Guide

XAttr Extension for Novell® Storage Services™ (NSS) makes it easy to back up and restore the NSS file metadata that is not currently exposed by other UNIX\* functions.

This feature is not supported on the existing NetWare® file system, but works only with the NSS environment that is included in Open Enterprise Server for Linux SP3.

This guide contains the following sections:

- [Chapter 1, “Concepts,” on page 9](#)
- [Chapter 2, “Functions,” on page 11](#)
- [Chapter 3, “Structures,” on page 17](#)
- [Chapter 4, “Values,” on page 21](#)
- [Appendix A, “Revision History,” on page 25](#)

## Audience

This guide is intended for NSS UNIX and Linux\* developers who want to read and set the extended metadata attributes for backup and restore purposes.

## Feedback

We want to hear your comments and suggestions about this manual and the other documentation included with this product. Please use the User Comments feature at the bottom of each page of the online documentation.

## Documentation Updates

We intend to post this document on the [Novell Developer Kit Web site \(http://developer.novell.com/ndk\)](http://developer.novell.com/ndk) in the near future.

## Additional Information

For the related developer support postings for Novell Storage Services, see the entries for File System Services (64-Bit) on the [Developer Support Forums \(http://developer.novell.com/ndk/devforums.htm\)](http://developer.novell.com/ndk/devforums.htm) Web site.

## Documentation Conventions

In Novell documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

A trademark symbol (® , ™ , etc.) denotes a Novell trademark. An asterisk (\*) denotes a third-party trademark.



# Concepts

# 1

Currently, Novell® Storage Services™ (NSS) stores some file metadata in Open Enterprise Server for Linux that is not accessible with common UNIX functions. The Target Service Agent (TSA) for NSS on Linux uses a special message interface to access the additional metadata. However, with the XAttr Extension included in SP3, you can now read, back up, and restore this file metadata without converting existing Linux products.

Future NSS releases will remain backward compatible with this feature.

This section covers the following topics:

- [Section 1.1, “Overview,” on page 9](#)
- [Section 1.2, “Data Sizes,” on page 9](#)
- [Section 1.3, “CtimeIsMetadataModTime Command Line Option,” on page 10](#)

XAttr Extension presents extra metadata as an uninterpreted data blob to any backup applications. The same data blob can then be passed to `setxattr` to restore the metadata, while preserving all the backed-up time stamps for the file.

For sample test files that use this functionality, see the `nw_metadata.c`, `libmetadata.c`, and `restore.c` files that are included with this documentation.

## 1.1 Overview

XAttr Extension uses the common Linux `getxattr` and `setxattr` functions and adds a new `netware.metadata` extended attribute to access additional file metadata that is not available by using other command line options.

Basically, you call `getxattr` to read your file attributes and one of its parameters points to the `zNW_metadata_s` ([page 18](#)) structure that returns the extended metadata. You can then save this data to a backup location.

To restore or set the extended metadata, call `setxattr`, specifying which fields you want to modify.

By calling these two functions, you can read, modify, back up, and restore all the metadata for your file. For more information on using `getxattr` and `setxattr`, see [Chapter 2, “Functions,” on page 11](#).

## 1.2 Data Sizes

Be aware that XAttr is limited to a total data size of 64 KB.

The size of the data returned by the `netware.metadata` attribute is defined as follows:

Key Word	Size
WORD	u16
LONG	u32
QUAD	u64

---

Key Word	Size
SQUAD	s64
GUID_t	u128

---

## 1.3 CtimeIsMetadataModTime Command Line Option

Currently, the Linux Semantic Agent (LSA) uses the creation time as the value for ctime. We'd like it to use the modified metadata time as its value, but there might be unknown complications from modifying this value. Instead, we've provided a new NSS command line CtimeIsMetadataModTime option.

This option allows an administrator to select that the ctime value should be the modified metadata time rather than the creation time.

For more information on how to use NSS command line commands, see [Using NSS Server Console Commands and Utilities \(http://www.novell.com/documentation/oes/nss\\_enu/data/ajhvfjd.html\)](http://www.novell.com/documentation/oes/nss_enu/data/ajhvfjd.html) in the *Novell Storage Services File System Administration Guide for OES*.

The CtimeIsMetadataModTime option is set in `/opt/novell/nss/conf/nssstart.cfg` and can be set from nsscon, as follows:

```
nss/ (no) CtimeIsMetadataModTime
```

# Functions

# 2

The following functions are not supported on the existing NetWare<sup>®</sup> file system but work only with the backup and restore environment that is included in OES for Linux SP3:

- “getxattr” on page 12
- “listxattr” on page 13
- “removexattr” on page 14
- “setxattr” on page 15

# getxattr

Reads a file's attributes and metadata.

## Syntax

```
ssize_t getxattr (  
    const char *path,  
    const char *name,  
    void *value,  
    size_t size  
);
```

## Parameters

### path

In OES, this parameter is named *file* and points to the name of the file to read attributes from.

### name

In OES, this parameter is named *xattr* and points to the [zNW\\_metadata\\_s \(page 18\)](#) structure.

### value

In OES, this parameter is named *before* and is a pointer to a location within your file from which to start reading.

### size

In OES, this parameter is named *sizeof* and specifies how much data has been read.

# listxattr

Lists a file's extended attribute names.

## Syntax

```
ssize_t listxattr (  
    const char *path,  
    char *list,  
    size_t size  
);
```

## Parameters

### path

Points to the name of the file to read extended attributes from.

### list

Points to the list of extended attributes.

### size

Specifies the length of the attribute list.

## Remarks

Extended attributes are name value pairs that are associated with files, directories, symbolic links, etc. They are extensions to the normal attributes that are associated with all files in the system.

listxattr retrieves a list of NULL-terminated extended attribute names that are associated with the given path in the filesystem.

listxattr also returns network metadata for every file. Utilities can call listxattr to retrieve the metadata necessary for backing up files. However, this is an optional feature which is turned off by default. To turn on this feature, use `nss/ListXattrNWmetadata` in `nsscon` or make it persistent by putting in `nssstart.conf`.

## removexattr

Removes the specified extended attribute from a file.

### Syntax

```
int removexattr (  
    const char *path,  
    const char *name  
);
```

### Parameters

#### path

Points to the name of the file to remove the extended attribute from.

#### name

Points to the extended attribute to remove.

### Return Values

On success, returns zero. On failure, returns -1 and sets errno.

### Remarks

Extended attributes are name/value pairs that are associated with files, directories, symbolic links, etc. They are extensions to the normal attributes associated with all files in the system.

An extended attribute name is a simple NULL-terminated string. The name includes a namespace prefix. There might be several namespaces associated with an individual extended attribute.

---

**IMPORTANT:** This function does not remove the `netware.metadata` extended attribute, but it does remove all the trustees and directory space restrictions associated with the file or directory.

---

## setxattr

Modifies a file's attributes and metadata.

### Syntax

```
ssize_t setxattr (  
    const char *path,  
    const char *name,  
    void *value,  
    size_t size,  
    int flags  
);
```

### Parameters

#### path

In OES, this parameter is named *file* and points to the name of the file to modify attributes in.

#### name

In OES, this parameter is named *xattr* and points to the [zNW\\_metadata\\_s \(page 18\)](#) structure.

#### value

In OES, this parameter is named *before* and is a pointer to a location within your file from which to start reading on subsequent calls.

#### size

In OES, this parameter is named *sizeof* and specifies how much data has been modified.

#### flags

Specifies what data to modify.



# Structures

# 3

The following structures work with the backup and restore environment that is included in OES for Linux SP3:

- “zNW\_metadata\_s” on page 18
- “zNW\_trustee\_s” on page 20

## zNW\_metadata\_s

Contains the extended file metadata.

**Defined In:** public/zXattr.h

### Syntax

```
typedef struct zNW_metadata_s {
    WORD                nwm_byteorder;
    WORD                nwm_version;
    LONG                nwm_reserved_1;
    QUAD                nwm_modify_mask;
    QUAD                nwm_file_attributes;
    QUAD                nwm_file_attributes_mask;

    QUAD                nwm_time_created;
    QUAD                nwm_time_archived;
    QUAD                nwm_time_modified;
    QUAD                nwm_time_accessed;

    QUAD                nwm_time_meta_data_modified;
    QUAD                nwm_reserved_2;
    GUID_t              nwm_id_owner;

    GUID_t              nwm_id_archiver;
    GUID_t              nwm_id_modifier;

    GUID_t              nwm_id_metadata_modifier;
    SQUAD               nwm_quota_limit;
    LONG                nwm_inherited_rights_mask;
    LONG                nwm_trustee_num;

    zNW_trstuee_s      nwm_trustee[ZMAX_TRUSTEES];
} zNW_metadata_s;
```

### Fields

#### nwm\_byteorder

Specifies the byte order of the structure.

#### nwm\_version

Specifies the version of the structure.

#### nwm\_reserved\_1

Is reserved. Set to 0.

#### nwm\_modify\_mask

Specifies the attributes (and corresponding fields in this structure) to modify. For possible values, see [Section 4.2, “Modify Mask Values,” on page 22](#).

#### nwm\_file\_attributes

Specifies the NSS file attributes. For possible values, see [Section 4.1, “File Attribute Values,” on page 21](#).

**nwm\_file\_attributes\_mask**

Specifies the NSS file attributes to modify.

**nwm\_time\_created**

Specifies the UTC time when the file was created.

**nwm\_time\_archived**

Specifies the UTC time when the file was last archived.

**nwm\_time\_modified**

Specifies the UTC time when the file was last modified.

**nwm\_time\_accessed**

Is no longer used. Currently, it is set to zero and `nwm_modify_mask` clears the `zMOD_ACCESSED_TIME` bit.

**nwm\_time\_meta\_data\_modified**

Specifies the UTC time when the file’s metadata was last modified.

**nwm\_reserved\_2**

Is reserved. Set to 0.

**nwm\_id\_owner**

Specifies the ID of the person who currently owns the file.

**nwm\_id\_archiver**

Specifies the ID of the person who last archived the file.

**nwm\_id\_modifier**

Specifies the ID of the person who last modified the file.

**nwm\_id\_metadata\_modifier**

Specifies the ID of the person who last modified the file’s metadata.

**nwm\_quota\_limit**

Specifies the quota limit that is currently assigned to the file.

**nwm\_inherited\_rights\_mask**

Specifies the inherited rights to return for the file.

**nwm\_trustee\_num**

Specifies the number of the file’s trustee.

**nwm\_trustee**

Points to the `zNW_trustee_s` structure, which contains trustee rights and ID information.

## **zNW\_trustee\_s**

Contains the rights and ID information for trustees.

**Defined In:** public/zXattr.h

### **Syntax**

```
typedef struct zNW_trustee_s {
    GUID_t    nwt_id;
    QUAD      nwt_rights;
    QUAD      nwt_reserved_2;
} zNW_trustee_s;
```

### **Fields**

#### **nwt\_id**

Specifies the trustee ID.

#### **nwt\_rights**

Specifies the rights for the given trustee.

#### **nwt\_reserved\_2**

Is reserved. Set to 0.

### **Remarks**

You can control the total size of `zNW_trustee_s` by defining `zMAX_TRUSTEES` before you include the `public/zXattr.h` file.

In order for this structure to fit in the 64 KB size limitation, we picked 2043 bytes as the maximum size of this structure, as the following snippet (from the `public/zXattr.h` file) shows:

```
#ifndef zMAX_TRUSTEES
#define zMAX_TRUSTEES 2043
#endif
```

# Values

# 4

The following values are defined for the backup and restore environment that is included in OES for Linux SP3:

- [Section 4.1, “File Attribute Values,” on page 21](#)
- [Section 4.2, “Modify Mask Values,” on page 22](#)

## 4.1 File Attribute Values

The following values are defined for the `nwm_file_attributes` field in the `zNW_metadata_s` (page 18) structure:

**Table 4-1** File Attribute Values

Value	Name	Description
0x00000001	zFA_READ_ONLY	Read only file.
0x00000002	zFA_HIDDEN	Hidden file.
0x00000004	zFA_SYSTEM	System file.
0x00000008	zFA_EXECUTE	Executable file.
0x00000010	zFA_SUBDIRECTORY	
0x00000020	zFA_ARCHIVE	Archiveable file.
0x00000080	zFA_SHAREABLE	Shareable file.
0x00000700	zFA_SMODE_BITS	
0x00000800	zFA_NO_SUBALLOC	
0x00001000	zFA_TRANSACTION	
0x00002000	zFA_NOT_VIRTUAL_FILE	Valid only on a volume with the <code>zATTR_VIRTUAL_FILES</code> attribute.
0x00010000	zFA_IMMEDIATE_PURGE	
0x00020000	zFA_RENAME_INHIBIT	
0x00040000	zFA_DELETE_INHIBIT	
0x00080000	zFA_COPY_INHIBIT	
0x00100000	zFA_IS_ADMIN_LINK	The file contains persistent administrator link information.
0x00200000	zFA_IS_LINK	
0x00400000	zFA_REMOTE_DATA_ACCESS	File was migrated.
0x00800000	zFA_REMOTE_DATA_INHIBIT	File won't allow data migration.

Value	Name	Description
0x02000000	zFA_COMPRESS_FILE_IMMEDIATELY	
0x04000000	zFA_DATA_STREAM_IS_COMPRESSED	Set per data stream directory entry.
0x08000000	zFA_DO_NOT_COMPRESS_FILE	
0x20000000	zFA_CANT_COMPRESS_DATA_STREAM	No space is saved by compressing this file.
0x40000000	zFA_ATTR_ARCHIVE	
0x80000000	zFA_VOLATILE	Data is volatile (no oplocks).

## 4.2 Modify Mask Values

The following values are defined for the `nwm_modify_mask` field in the `zNW_metadata_s` (page 18) structure:

**Table 4-2** *Modify Mask Values*

Value	Name	Description
0x00000001	zMOD_FILE_ATTRIBUTES	
0x00000002	zMOD_CREATED_TIME	
0x00000004	zMOD_ARCHIVED_TIME	
0x00000008	zMOD_MODIFIED_TIME	
0x00000010	zMOD_ACCESSED_TIME	This bit is no longer used. Instead, the <code>nwm_time_accessed</code> field is set to zero and this bit is cleared.
0x00000020	zMOD_METADATA_MODIFIED_TIME	
0x00000040	zMOD_OWNER_ID	
0x00000080	zMOD_ARCHIVER_ID	
0x00000100	zMOD_MODIFIER_ID	
0x00000200	zMOD_METADATA_MODIFIER_ID	
0x00000400	zMOD_PRIMARY_NAMESPACE	If the volume is mounted in the LONG namespace, no access is provided to UNIX, Mac, or DOS namespaces. However, because the alternate namespaces are invisible, the generated alternate names are sufficient.
0x00000800	zMOD_DELETED_INFO	
0x00001000	zMOD_MAC_METADATA	
0x00002000	zMOD_UNIX_METADATA	
0x00004000	zMOD_EXTATTR_FLAGS	

<b>Value</b>	<b>Name</b>	<b>Description</b>
0x00008000	zMOD_VOL_ATTRIBUTES	
0x00010000	zMOD_VOL_NDS_OBJECT_ID	
0x00020000	zMOD_VOL_MIN_KEEP_SECONDS	
0x00040000	zMOD_VOL_MAX_KEEP_SECONDS	
0x00080000	zMOD_VOL_LOW_WATER_MARK	
0x00100000	zMOD_VOL_HIGH_WATER_MARK	
0x00200000	zMOD_POOL_ATTRIBUTES	
0x00400000	zMOD_POOL_NDS_OBJECT_ID	
0x00800000	zMOD_VOL_DATA_SHREDDING_COUNT	
0x01000000	zMOD_VOL_QUOTA	
0x02000000	zMOD_DIR_QUOTA	
0x04000000	zMOD_READ_AHEAD_BLOCKS	
0x08000000	zMOD_INH_RIGHTS_MASK	
0x10000000	zMOD_ALL_TRUSTEES	



# Revision History

# A

The following table outlines all the changes that have been made to the XAttr Extension for Novell® Storage Services™ documentation (in reverse chronological order).

---

October 11, 2006	Updated <a href="#">zNW_metadata_s (page 18)</a> to show that the <code>nwm_time_accessed</code> field is no longer used. Instead, it is set to zero and <code>nwm_modify_mask</code> clears the <code>zMOD_ACCESSED_TIME</code> bit.
June 21, 2006	Added <a href="#">listxattr (page 13)</a> and <a href="#">removexattr (page 14)</a> .
March 1, 2006	Added as a new NDK component.
January 23, 2006	Created as a new document.

---