CLAIMS:

1. A method of operating a policy aware unified file system of a computer system comprising:

unifying existing branches in the computer system to create the unified file system wherein the unified file system includes branch policies of the unified existing branches and a unification policy of the unified file system, and wherein the branch policies of the unified existing branches include security policies of the unified existing branches;

receiving a request from an application to access the unified file system;

determining if the request satisfies the unification policy of the unified file system and a corresponding branch policy;

if the request satisfies the unification policy and the corresponding branch policy, allowing the application to access the unified file system;

wherein the method further comprises a lazy creation optimization process when a software application accesses a file in the unified file system:

determining if the request relates to an access of an existing file in the unified file system;

if the request does not relate to the access of an existing file, determining if the file exists in the branch file system;

if the file is determined to exist in the branch file system, determining if unifying the file into the unified file system satisfies both unification policy and branch policy; and

if unifying the file into unified file system satisfies both unification policy and branch policy, creating a virtual file in the unified file system for the file.

2. The method according to claim 1, wherein the unifying the existing branches in the computer system to create the unified file system comprises:

a) selecting a branch from the existing branches in the computer system;
b) retrieving a branch policy associated with the selected branch and retrieving a unification policy associated with the unified file system;
c) creating virtual files in the unified file system for all the files in the selected branch wherein each created virtual file corresponds to a file in the selected branch;
d) applying the retrieved branch policy associated with the selected branch and the retrieved unification policy to all the virtual files created for the files in the selected branch; and
e) repeating steps (a) to (d) until all branches in the computer system have been selected.

3. The method according to claim 1, wherein the unifying the existing branches in the computer system to create the unified file system comprises:
    a) selecting a branch from the existing branches in the computer system;
    b) retrieving a branch policy associated with the selected branch and retrieving a unification policy associated with the unified file system;
    c) selecting a file from the branch;
    d) determining if the selected file satisfies the retrieved branch policy associated with the selected branch and satisfies the retrieved unification policy;
    e) if the selected file satisfies the retrieved branch policy associated with the selected branch and the retrieved unification policy, creating a virtual file in the unified file system for the selected file;
    f) repeating steps (c) to (e) until all the files in the selected branch have been selected;
    g) repeating steps (a) to (f) until all the branches in the computer system have been selected.

4. The method according to claim 1, wherein the unified file system is created in a different location within the computer system from the locations of the existing branches in the computer system.

5. The method according to claim 3, wherein the creating the virtual file in the unified file system for the selected file further comprises:
    creating a symbolic link or a dummy file to the selected file;
    adding the symbolic link or the dummy file to the unified file system; and
    updating a metadata of the unified file system, wherein a link relationship between the symbolic link or the dummy file with the selected file is recorded in the metadata.

6. The method according to claim 1 further comprising:
    receiving from the application a query request to query link relationships of virtual files in the unified file system;
    using metadata of the unified file system to obtain the link relationships between the files in a first file system and their corresponding virtual files in the unified file system; and
    providing the link relationships to the application.

7. The method according to claim 1 further comprising:
    determining if the request relates to an access of an existing file in the unified file system;
    if the request relates to the access of an existing file in the unified file system:

determining if the request to access an existing file satisfies the unification policy and a branch policy associated with the existing file to be accessed; and

if the request to access an existing file satisfies the unification policy and the branch policy of the existing file to be accessed allowing the application to access the existing file within the unified file system.

determining if the request to access the existing file is a deletion request;

If the request to access the existing file is determined to be the deletion request, updating a metadata to delete a link relationship between the existing file and a corresponding virtual file.

8. The method according to claim 1 further comprising:

determining if the request relates to an addition of a new file to the unified file system: if the request relates to the addition of the new file to the unified file system:

obtaining from the unification policy a branch the new file is to be added to;

determining if the addition of the new file satisfies the unification policy and a branch policy of the branch the new file is to be added to, and if the addition of the new file satisfies the unification policy and the branch policy of the branch the new file is to be added to:

adding the new file to the branch;

creating a virtual file in the unified file system for the new file added to the branch; and

applying the branch policy of the branch and the unification policy to the virtual file.

9. A system for operating a policy aware unified file system of a computer system, the system comprising:

a processing unit; and

a non-transitory media readable by the processing unit, the media storing instructions that when executed by the processing unit cause the processing unit to:

unify existing branches in the computer system to create the unified file system, wherein the unified file system includes branch policies of the unified existing branches and a unification policy of the unified file system, and wherein the branch policies of the unified existing branches include security policies of the unified existing branches;

receive a request from an application to access the unified file system;

determine if the request satisfies the unification policy of the unified file system and a corresponding branch policy;

if the request satisfies the unification policy and the corresponding branch policy, to allow the application to access the unified file system;

wherein the method further comprises a lazy creation optimization process when a software application accesses a file in the unified file system:

determining if the request relates to an access of an existing file in the unified file system;

if the request does not relate to the access of an existing file, determining if the file exists in the branch file system;

if the file is determined to exist in the branch file system, determining if unifying the file into the unified file system satisfies both unification policy and branch policy; and

if unifying the file into unified file system satisfies both unification policy and branch policy, creating a virtual file in the unified file system for the file.

10. The system according to claim 9, wherein the instructions to unify the existing branches in the computer system to create the unified file system comprises:

instructions for directing the processing unit to:

a) select a branch from the existing branches in the computer system;
b) retrieve a branch policy associated with the selected branch and retrieve a unification policy associated with the unified file system;
c) create virtual files in the unified file system for all the files in the selected branch wherein each created virtual file corresponds to a file in the selected branch;
d) apply the retrieved branch policy associated with the selected branch and the retrieved unification policy to all the virtual files created for the files in the selected branch; and
e) repeat steps (a) to (d) until all branches in the computer system have been selected.

11. The system according to claim 9, wherein the instructions to unify the existing branches in the computer system to create the unified file system comprises:

instructions for directing the processing unit to:

a) select a branch from the existing branches in the computer system;
b) retrieve a branch policy associated with the selected branch and retrieve a unification policy associated with the unified file system;
c) select a file from the branch;
d) determine if the selected file satisfies the retrieved branch policy associated with the selected branch and satisfies the retrieved unification policy;

e) if the selected file satisfies the retrieved branch policy associated with the selected branch and the retrieved unification policy, to create a virtual file in the unified file system for the selected file;

f) repeat steps (c) to (e) until all the files in the selected branch have been selected;

g) repeat steps (a) to (f) until all the branches in the computer system have been selected.

12. The system according to claim 9, wherein the unified file system is created in a different location within the computer system from the locations of the existing branches in the computer system.

13. The system according to claim 10 , wherein the instructions to create the virtual file in the unified file system for the selected file further comprises:

instructions for directing the processing unit to:

create a symbolic link or a dummy file to the selected file;

add the symbolic link or the dummy file to the unified file system; and

update a metadata of the unified file system, wherein a link relationship between the symbolic link or the dummy file with the selected file is recorded in the metadata.

14. The system according to claim 9 further comprising:

instructions for directing the processing unit to:

receive from the application a query request to query link relationships of virtual files in the unified file system;

use metadata of the unified file system to obtain the link relationships between the files in a first file system and their corresponding virtual files in the unified file system; and

provide the link relationships to the application.

15. The system according to claim 9 further comprising:

instructions for directing the processing unit to:

determine if the request relates to an access of an existing file in the unified file system;

if the request relates to the access of an existing file in the unified file system:

determine if the request to access an existing file satisfies the unification policy and a branch policy associated with the existing file to be accessed; and

if the request to access an existing file satisfies the unification policy and the branch policy of the existing file to be accessed, to allow the application to access the existing file within the unified file system.

determine if the request to access the existing file is a deletion request;

if the request to access the existing file is determined to be the deletion request, to update a metadata to delete a link relationship between the existing file and a corresponding virtual file.

16. The system according to claim 9 further comprising:

instructions for directing the processing unit to:

determine if the request relates to an addition of a new file to the unified file system:

if the request relates to the addition of the new file to the unified file system:

obtain from the unification policy a branch the new file is to be added to;

determine if the addition of the new file satisfies the unification policy and a branch policy of the branch the new file is to be added to, and if the addition of the new file satisfies the unification policy and the branch policy of the branch the new file is to be added to:

add the new file to the branch;

create a virtual file in the unified file system for the new file added to the branch; and

apply the branch policy of the branch and the unification policy to the virtual file.