

Blacklist Transfer Manager

- **Introduced in:**
- **Contract name:** BlacklistTransferManager.sol
- **Type:** Transfer Manager Module
- **Compatible Protocol Version:** TBD
- **Associated LucidChart:**
<https://www.lucidchart.com/documents/edit/97d5e109-2948-4916-b86f-a32521e359bd/0?shared=true&>

How it works

This module will allow approved staff to create and establish an automated blacklist to prevent Insiders from selling thus de-risking price swings during time sensitive cases like earning calls. The blacklists can automatically become active/inactive at set recurring intervals.

Key functionalities (as defined in the Smart Contract)

Initialization

This module does not require any initialization.

Transfer Verification

- If the module is paused or no blacklist is imposed on the sender, Transfer restrictions are skipped for that transfer.
- The module parses through all the blacklists applied on the sender and returns invalid transfer if any of the blacklist is currently active.
- Transfer are allowed to go through if none of the blacklists applied on the sender are active at the moment.

Managing blacklist types

Admins can add a new blacklist, modify an existing blacklist or delete a blacklist at any time using the below functions:

```
function addBlacklistType(uint256 _startTime, uint256 _endTime, bytes32 _name, uint256 _repeatPeriodTime) public withPerm(ADMIN)
```

is used to add a new blacklist type.

```
function addBlacklistTypeMulti(uint256[] _startTimes, uint256[] _endTimes, bytes32[] _names, uint256[] _repeatPeriodTimes) public withPerm(ADMIN)
```

is used to add the multiple blacklist type in a one transaction and it have the same permission as `addBlackListType()` has.

```
function modifyBlacklistType(uint256 _startTime, uint256 _endTime, bytes32 _name, uint256 _repeatPeriodTime) public withPerm(ADMIN)
```

is used to modify an existing blacklist type.

```
function modifyBlacklistTypeMulti(uint256[] _startTimes, uint256[] _endTimes, bytes32[] _names, uint256[] _repeatPeriodTimes) public withPerm(ADMIN)
```

is used to modify multiple existing blacklist types and it have the same access permission as `modifyBlacklistType()` has

```
function deleteBlacklistType(bytes32 _name) public withPerm(ADMIN)
```

is used to delete a blacklist type.

```
function deleteBlacklistTypeMulti(bytes32[] _names) public withPerm(ADMIN)
```

is used to delete the multiple blacklist types in a transaction.

`_startTime` is the timestamp of the beginning of the first instance of the blacklist.

`_endTime` is the timestamp of the end of the first instance of the blacklist.

`_repeatPeriodTime` is number of cooldown days after which the blacklist should be re activated. If the `_repeatPeriodTime == 0` makes the period non-repeating.

`_name` is the name of the blacklist type.

Applying blacklist to investors

Admins can add or remove users from blacklists at any time using the below functions:

```
function addInvestorToBlacklist(address _investor, bytes32
_blacklistName) public withPerm(ADMIN)
```

is used to apply a specific blacklist to a investor. `_blacklistName` should be pre-exist in the contract storage otherwise transaction get failed.

```
function addInvestorToBlacklistMulti(address[] _investors, bytes32
_blacklistName) public withPerm(ADMIN)
```

is used to impose a blacklist on multiple investors.

```
function addMultiInvestorToBlacklistMulti(address[] _investors,
bytes32[] _blacklistNames) public withPerm(ADMIN)
```

is used to assign the multiple blacklist type to the multiple investor.

```
function addInvestorToNewBlacklist(uint256 _startTime, uint256
_endTime, bytes32 _name, uint256 _repeatPeriodTime, address _investor)
public withPerm(ADMIN)
```

is used to assign the new blacklist type to an investor, It will create a new blacklistType and assign to the investor.

```
function deleteInvestorFromBlacklist(address _investor, bytes32
_blacklistName) public withPerm(ADMIN)
```

is used to remove a specific blacklist from an investor.

```
function deleteInvestorFromAllBlacklist(address _investor) public
withPerm(ADMIN)
```

is used to remove all blacklists imposed on an investor.

```
function deleteInvestorFromAllBlacklistMulti(address[] _investors)
public withPerm(ADMIN)
```

is used to delete the multiple investor from all the associated blacklist types.

```
function deleteMultiInvestorsFromBlacklistMulti(address[] _investors,
bytes32[] _blacklistNames) public withPerm(ADMIN)
```

is used to delete the multiple investor from the multiple blacklist.

`_investor` is the Ethereum address of the investor.
`_blacklistName` is the name of the blacklist.

Special considerations / notes

function `addInvestorToBlacklistMulti` is limited by GAS limit and can only process a maximum number of address at a time.

To set a non recurring blacklist, `_repeatPeriodTime` can be set to a large number.

All investors must be removed from a blacklist before that blacklist can be deleted using `deleteBlacklistType` function.

Troubleshooting / FAQs

None

Know Issues / bugs

None

Changelog

Fixed:

- Blacklist periods now begin x days after start time (and not end time as before).
This allows the repeat period to start from the start time rather than the end time. (Note: Having 0 repeat periods is allowed)
- 'addInvestorToBlacklist' now allows you to only add an investor once