

# It's **NOT** Mine! General Purpose API

## 1. Objective

To allow the seamless integration of, and interaction between, items using the General Purpose API and the It's **Not** Mine! Body Cum System to control the visual effect on the recipient, through their It's **NOT** Mine! Body Cum System. This can be achieved through the abilities of the API to present Cum or Lick Location menus to a user, or directly affecting layers of the recipients It's **NOT** Mine! Body Cum System without menus, often times bypassing most internal checks done by the It's **NOT** Mine! System, therefore requiring a strict licensing and license validation process to ensure the safety of users and prevent misuse of the General Purpose API.

## 2. Document history

Version	Date	Version comments
1.0		Initial version
1.1	11-17-2017	Undocumented Changes/Bug Fixes
1.2		Undocumented Changes/Bug Fixes
1.3	2-24-2019	
1.4		
1.5	1-25-2020	License validation URL correction, Changes to license validation process and requirements, Additional target areas added to support upcoming INM release
1.6	2-20-2020	Minor revision to re-add missing action commands
1.7	Good Question	Not sure, I forgot to update this doc.
1.8	9-30-2022	Additional commands added. See Appendix F.

### 3. Document jargons

Jargon	Meaning
SL	Second Life®
TOS	Second Life® Terms of Service
INM	It's <b>Not</b> Mine! Body Cum System
Avatar	An entity that represents a valid account in Second Life®
API	It's <b>Not</b> Mine! General Purpose API
Script	Any technical object in Second Life® that represents a program in LSL (Linden Scripting Language) or any other language
Object	Any object existing in Second Life®. In case the objects contain scripts, they will be called "scripted objects".
COPY	Second Life® parameter that indicates that any object can be copied by its the next owner
MOD	Second Life® parameter that indicates that any object can be modified by its the next owner
TRANSFER	Second Life® parameter that indicates that any object can be transferred from its the next owner to other parties

### 4. Disclaimer

The It's **Not** Mine! General Purpose API is distributed with no warranty by TrinaCarlsson Resident at her sole discretion. You are not authorized to distribute the API as copy and transfer simultaneously to any other party under any circumstance.

TrinaCarlsson Resident will not be held responsible for misuse of the API for purposes which are not condoned by the Second Life® TOS or Community Standards for damages infringed to other parties by any behavior caused by object or avatar, scripted or not, directly or indirectly controlled by another avatar, or automatons (scripts or scripted objects that will behave without the need of third party operation).

TrinaCarlsson Resident will not provide support for LSL scripting or do custom work

TrinaCarlsson Resident reserves the right to revoke the license of any creator that was previously licensed to use this API, thereby disabling and preventing the further use of the API, for any reason or time as it is deemed necessary.

## 5. System requirements

- A valid Second Life® account
- A valid Second Life® viewer
- Internet connection
- A copy of It's **Not** Mine! General Purpose API
- A valid license to use the API in your creations
- An object created by you, which will remain as the root prim for your completed creation and will contain the API.
- Knowledge of, and ability to interact with, scripts using LSL in order to send and receive commands to/from the API

## 6. Object and API requirements for successful license validation

- An object (prim or mesh) which was created by you and that you are the current owner of.
- API MUST be set to either **COPY** or **TRANSFER** only.
- Object permissions **MUST** be set to **NO MODIFY** for next owner

### **Note:**

*API license validation will fail if it detects that any of the above conditions have not been met! License validation will be required any time the API resets or object inventory changes.*

## 7. Limitations imposed on the use of this API

- Using the API in an attached item will prevent the API from affecting an avatar other than the owner of the attached object.
- The object this API is used in **MUST** be **NO MODIFY** for end users! Changes to the contents of the object containing the API will invalidate the API license. Validation can only occur while the object creator is also the object owner!
- Link messages sent by the API can only be heard by scripts in the prim that contains the API

## 8. API Commands and Syntax

The INM General Purpose API will only respond to linked messages being sent to it using the number value of 2550. All responses from the API will be sent using the number value of 2551.

### **Disable Automatic Cum Menus:**

This command disables the INM menu system and prevents it from displaying cum or lick menus when orgasm is detected in a supported genital product, or a lick event is triggered in the INM system. This value defaults to 0 whenever the INM system is attached or the user logs in wearing their INM system.

*Linked message to send to API:*

```
!!MessageLinked(LINK_THIS/SET, 2550, "DisableMenu|1", <target_av_key>);
```

**Target\_av\_key** is the UUID (key) for owner of the cum system you are disabling the menu in.

**Note:**

This command Does NOT prevent the API from triggering a menu and giving it to someone or making cum dis/appear on a user!

**Note:**

Regardless of the key sent, <target\_av\_key> will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

### **Re-enable Automatic Cum Menus:**

This command re-enables the INM menu system and allows it displaying cum and lick menus when orgasm is detected in a supported genital product, or a lick event is triggered in the INM system. This returns the system to its normal operation

*Linked message to send to API:*

```
//MessageLinked(LINK_THIS/SET, 2550, "DisableMenu|0", <target_av_key>);
```

Target\_av\_key is the UUID (key) for owner of the cum system you are re-enabling the menu in.

**Note:**

DO NOT FORGET TO SEND THIS COMMAND ONCE THE MENU SYSTEM NO LONGER REQUIRES BEING DISABLED!!

**Note:**

Regardless of the key sent, <target\_av\_key> will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

## **Trigger the Cum Location Menu**

This command will cause the target\_av's INM menu system to present the Cum location menu to the specified user.

*Linked message to send to API:*

```
//MessageLinked(LINK_THIS/SET, 2550, "Cum_Menu/" + <uuid you want the menu to go to>, <target_av_key>);
```

**uuid you want the menu to go to** is the avatar that the target\_av\_key INM system should send the Cum location menu to.

**Target\_av\_key** is the UUID (key) for owner of the cum system you want trigger the menu from.

**Note:**

Regardless of the key sent, **<target\_av\_key>** will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

## **Trigger the Lick Location Menu**

This command will cause the target\_av's INM menu system to present the Lick location menu to the specified user.

*Linked message to send to API:*

```
//MessageLinked(LINK_THIS/SET, 2550, "Lick_Menu|" + <uuid you want the menu to go to>, <target_av_key>);
```

**uuid you want the menu to go to** is the avatar that the target\_av\_key INM system should send the Lick location menu to.

**Target\_av\_key** is the UUID (key) for owner of the cum system you want to request the lick menu from.

**Note:**

Regardless of the key sent, **<target\_av\_key>** will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

### **Make cum appear on a user:**

This command will make one cum layer visible on the designated target area of the target\_av. Issuing commands through the use of the API will circumvent delay settings, and a number of other user related settings that might prevent menu generation or one's ability to cum on a specific area.

#### ***Linked message to send to API:***

```
llMessageLinked(LINK_THIS/SET, 2550, <target_area> + "|" + <uuid for avi/object that cums> + "|" + <update_stats>", <target_av_key>);
```

**target\_area** is the area of the target\_av\_key's body where the cum layer should be made visible. *See Appendix A for a complete list of target\_area variables*

**uuid for avi/object that cums** is the key for the avatar or item responsible for making the cum layer visible.

**Update\_stats** is a variable the INM system uses to determine if it should update a users personal statistics. Accepted values are 0 or 1.

Statistics updates will only be sent if an avatar within the same region as the user is listed as the uuid responsible for making the cum appear on the user!

**Target\_av\_key** is the UUID (key) for owner of the cum system you want to make cum appear on.

#### **Note:**

Regardless of the key sent, **<target\_av\_key>** will revert to the key for the owner of the object the API script is in if the object is a worn attachment!



### **Removing Cum from a specific area on a user:**

This command will remove one cum visible cum layer from the designated target area of the target\_av\_key. Issuing commands through the use of the API will circumvent delay settings, and a number of other user related settings that might prevent menu generation or one's ability to lick a specific area.

#### ***Linked message to send to API:***

```
llMessageLinked(LINK_THIS/SET, 2550, <target_area> + "|" + <uuid for avi/object that cums> + "|" + <update_stats>, <target_av_key>);
```

**target\_area** is the area of the target\_av\_key's body where the cum layer should be made visible.  
*See Appendix B for a complete list of target\_area variables*

**uuid for avi/object that cums** is the key for the avatar or item responsible for removing the cum layer.

**Update\_stats** is a variable the INM system uses to determine if it should update a users personal statistics. Accepted values are 0 or 1.

Statistics updates will only be sent if an avatar within the same region as the user is listed as the uuid responsible for making the cum appear on the user!

**Target\_av\_key** is the UUID (key) for owner of the cum system you want to remove cum from.

#### **Note:**

Regardless of the key sent, **<target\_av\_key>** will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

**Immediately remove all cum from a user (API bot version only):**

This command will remove all of the visible cum on the avatar designated as the target\_av\_key. The command is only available in the bot version of the API, specifically for creators utilizing and equipping bot avatars with INM systems.

***Linked message to send to API:***

```
//MessageLinked(LINK_THIS/SET, 2550, "Cleanup", <target_av_key>);
```

Target\_av\_key is the UUID (key) for owner of the cum system you want to remove cum from.

**Note:**

Regardless of the key sent, <target\_av\_key> will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

**Request a system status report from a users INM system:**

This command will request information from a users INM system and return the result of the request. The information provided from the request will be in the form of a pipe ("|") delimited link message and contain the current cum levels for the user, the object or avatar key responsible for the change in the system (if any), as well as the users current system settings.

**Linked message sent to API:**

```
llMessageLinked(LINK_THIS/SET, 2550, "Get_Status", <target_av_key>);
```

**Target\_av\_key** is the UUID (key) for owner of the cum system you requesting the status of.

The API will send a linked message upon receiving a response from the users system.

### Sample API response

Number: 2551

```
Msg: "Status|b0ad2885-c3e0-46ff-b828-27ae7e486812
|1.000000|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|0|
00000000-0000-0000-0000-000000000000|2063683543"
```

Key:

b0ad2885-c3e0-46ff-b828-27ae7e486812

**Note:**

The message does not report what changed, it only reports the current status of each value once a change is made. Your script will need to store and compare the last known values to the new values when they are received, in order to determine exactly what changed.

**Note:**

The API will send a Status update message, which is identical to the Get\_Status response message, any time something changes with the system, including cum layers being added or removed and during cleanup as the cum opacity changes.

Regardless of the key sent, **<target\_av\_key>** will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

## 9. Appendix A

### Supported Female Cum Target Areas

Target area name	Affected Body part	Add-on required	New in API version 1.4	Available in INM v3.2/INAM v2.3	Available in INM v3.3/INAM v2.4
On face	Face			✓	✓
On tits	Breasts			✓	✓
in mouth	In the mouth			✓	✓
R hand/arm	Right arm/right hand	✓		✓	✓
L hand/arm	Left arm/left hand	✓		✓	✓
Right leg	Right leg	✓		✓	✓
Left leg	Left leg	✓		✓	✓
On pussy	Pelvis and navel			✓	✓
In pussy	Pussy, visible as dripping			✓	✓
On ass	Ass and lower back			✓	✓
In ass	Ass, visible as dripping	✓		✓	✓
On crotch	Same as 'On Pussy' but used for She-male option			✓	✓
On neck	Above breasts, below head/neck seam		✓		✓
On stomach	Below breasts, above crotch		✓		✓
On back	Above butt, from shoulders down		✓		✓
On R Leg Bck	Back of right leg	✓	✓		✓
On L Leg Bck	Back of left leg	✓	✓		✓
On R Foot	Right foot, below ankle	✓	✓		✓
On L Foot	Left foot, below ankle	✓	✓		✓
On R Arm	Redundant, use "R Hand/arm"	✓	✓		✓
On L Arm	Redundant, use "L Hand/arm"	✓	✓		✓
On R Hand	Redundant, use "L Hand/arm"	✓	✓		✓
On L Hand	Redundant, use "L Hand/arm"	✓	✓		✓
On R Leg Frt	Redundant, use "Right Leg"	✓	✓		✓
On L Leg Frt	Redundant, use "Left Leg"	✓	✓		✓

## Supported Male Cum Target Areas

Target area name	Affected Body part	Add-on required	New in API version 1.7	Available in INM v3.2/INAM v2.3	Available in INM v3.3/INAM v2.4
On face	Face			✓	✓
On chest	chest			✓	✓
in mouth	In the mouth			✓	✓
R hand/arm	Right arm/right hand	✓		✓	✓
L hand/arm	Left arm/left hand	✓		✓	✓
Right leg	Right leg	✓		✓	✓
Left leg	Left leg	✓		✓	✓
On crotch	Pelvis and navel			✓	✓
On ass	Ass and lower back			✓	✓
In ass	Ass, visible as dripping			✓	✓
On Neck	Above pecs, below head/neck seam		✓		✓
On Back	Back, above butt from shoulders down		✓		✓
On Stomach	Below pecs, above crotch		✓		✓
On R Leg Bck	Back of Right Leg	✓	✓		✓
On L Leg Bck	Back of Left Leg	✓	✓		✓
On R Foot	Below right ankle	✓	✓		✓
On L Foot	Below Left ankle	✓	✓		✓
On R Leg Frt	Redundant, use "Right Leg"	✓	✓		✓
On L Leg Frt	Redundant, use "Left Leg"	✓	✓		✓
On R Arm	Redundant, use "R Hand/arm"	✓	✓		✓
On L Arm	Redundant, use "L Hand/arm"	✓	✓		✓
On R Hand	Redundant, use "R Hand/arm"	✓	✓		✓
On L Hand	Redundant, use "L Hand/arm"	✓	✓		✓

**Note:** Target areas marked in **RED** are duplicate commands!

*In order to maintain backwards compatibility, it is recommended that you continue to use the old target name when duplicates are provided!*

*Target areas not yet available in the INM/INAM system can still be configured in the API. The INM systems will simply ignore them until INM 3.3 and INAM v2.4 is released*

## 10. Appendix B

### Supported Female Lick Target Areas

Target area name	Affected Body part	Add-on required	New in API version 1.7	Available in INM v3.2/INAM v2.3	Available in INM v3.3/INAM v2.4
Lick face	Removes from face			✓	✓
Lick tits	Removes from breasts			✓	✓
Cum swap	Removes from mouth			✓	✓
Lick r arm	Removes from Right arm/right hand	✓		✓	✓
Lick l arm	Removes from Left arm/left hand	✓		✓	✓
Lick r leg	Removes from Right leg	✓		✓	✓
Lick l leg	Removes from Left leg	✓		✓	✓
Lick pussy	Removes from Pelvis and navel			✓	✓
Suck pussy	Removes visible dripping from pussy			✓	✓
Lick ass	Removes from Ass and lower back			✓	✓
Suck ass	Removes visible dripping from ass	✓		✓	✓
Lick crotch	Same as 'Lick pussy'			✓	✓
Lick Neck	Removes from neck area, above breasts, below head/neck seam		✓		✓
Lick Stomach	Removes from below breasts, above crotch		✓		✓
Lick Back	Removes from back area, above butt		✓		✓
Lick R Foot	Removes from below right ankle	✓	✓		✓
Lick L Foot	Removes from below left ankle	✓	✓		✓
Lick R Leg B	Removes from back of Right Leg	✓	✓		✓
Lick L Leg B	Removes from back of Left Leg	✓	✓		✓
Lick R Hand	Redundant, use "lick r arm"	✓	✓		✓
Lick L Hand	Redundant, use "lick l arm"	✓	✓		✓
Lick R Leg F	Redundant, use "Lick R Leg"	✓	✓		✓
Lick L Leg F	Redundant, use "Lick L Leg"	✓	✓		✓

## Supported Male Lick Target Areas

Target area name	Affected Body part	Add-on required	New in API version 1.7	Available in INM v3.2/INAM v2.3	Available in INM v3.3/INAM v2.4
Lick face	Removes from face			✓	✓
Lick chest	Removes from chest			✓	✓
Cum swap	Removes from mouth			✓	✓
Lick r arm	Removes from Right arm/right hand	✓		✓	✓
Lick l arm	Removes from Left arm/left hand	✓		✓	✓
Lick r leg	Removes from Right leg	✓		✓	✓
Lick l leg	Removes from Left leg	✓		✓	✓
Lick crotch	Removes from Pelvis and navel			✓	✓
Lick ass	Removes from Ass and lower back			✓	✓
Suck ass	Removes visible dripping from ass			✓	✓
Lick Neck	Removes from neck area, above pecs, below head/neck seam		✓		✓
Lick Stomach	Removes from below pecs, above crotch		✓		✓
Lick Back	Removes from back area, above butt		✓		✓
Lick R Foot	Removes from below right ankle	✓	✓		✓
Lick L Foot	Removes from below left ankle	✓	✓		✓
Lick R Leg B	Removes from back of Right Leg	✓	✓		✓
Lick L Leg B	Removes from back of Left Leg	✓	✓		✓
Lick R Leg F	Redundant, use "Lick R Leg"	✓	✓		✓
Lick L Leg F	Redundant, use "Lick L Leg"	✓	✓		✓
Lick R Hand	Redundant, use "lick r arm"	✓	✓		✓
Lick L Hand	Redundant, use "lick l arm"	✓	✓		✓

**Note:** Target areas marked in **RED** are duplicate commands!

*In order to maintain backwards compatibility, it is recommended that you continue to use the old target name when duplicates are provided!*

*Target areas not yet available in the INM/INAM system can still be configured in the API. The INM systems will simply ignore them until INM 3.3 and INAM v2.4 is released*

## 11. Appendix C

### Supported Self Action Commands

Target area name	Affected Body part	Add-on required	New in API version 1.7	Available in INM v3.2/INAM v2.3	Available in INM v3.3/INAM v2.4
Dribble	Moves cum from mouth to chest		✓		✓
Wipe Face	Causes user to use right hand to wipe cum from their face	✓	✓		✓
Spit	Removes from mouth		✓		✓
Swallow	Removes from mouth			✓	✓

These commands follow the same format as making cum appear or removing cum from a user:

```
!!MessageLinked(LINK_THIS/SET, 2550, <command> + "/" + <uuid for avi/object that cums> + "/" + <update_stats>, <target_av_key>);
```

**Note:**

Regardless of the key sent, *<target\_av\_key>* will revert to the key for the owner of the object the API script is in if the object is a worn attachment!



## 12. Appendix D

### Get\_Status Request response breakdown

```
link_message(integer link, integer num, string msg, key id) {  
  
    list data = IIParseStringKeepNulls(msg, ["|"], []);  
  
    string command = IIList2String(data, 0);  
  
    float cum_opacity = (float)IIList2String(data, 1);  
  
    integer cum_mouth = (integer)IIList2String(data, 2);  
  
    integer cum_face = (integer)IIList2String(data, 3);  
  
    integer cum_tits/chest = (integer)IIList2String(data, 4);  
  
    integer cum_pussy/crotch = (integer)IIList2String(data, 5);  
  
    integer cum_ass = (integer)IIList2String(data, 6);  
  
    integer cum_in_pussy = (integer)IIList2String(data, 7);  
  
    integer cum_in_ass = (integer)IIList2String(data, 8);  
  
    integer cum_r_arm = (integer)IIList2String(data, 9);  
  
    integer cum_r_leg_frt = (integer)IIList2String(data, 10);  
  
    integer cum_l_arm = (integer)IIList2String(data, 11);  
  
    integer cum_l_leg_frt = (integer)IIList2String(data, 12);  
  
    integer cum_neck = (integer)IIList2String(data, 13);  
  
    integer cum_back = (integer)IIList2String(data, 14);  
  
    integer cum_stomach = (integer)IIList2String(data, 15);  
  
    integer cum_r_leg_bck = (integer)IIList2String(data, 16);  
  
    integer cum_l_leg_bck = (integer)IIList2String(data, 17);  
  
    integer cum_r_foot = (integer)IIList2String(data, 18);  
  
    integer cum_l_foot = (integer)IIList2String(data, 19);  
  
    integer cum_r_hand = (integer)IIList2String(data, 20);  
  
    integer cum_l_hand = (integer)IIList2String(data, 21);  
  
    key cummer_avi/object = (key)IIList2String(data, -2);  
  
    integer system_settings = (integer)IIList2String(data, -1);  
  
    key Respoding_INM_System_Owner = (key)id;  
  
}
```

## 13. Appendix E

### System Settings variable breakdown

System Setting	Relevant bit in Sys_Settings Variable	Integer equivalent
System is Active	$1 \ll 0$	1
System is Detachable when clean	$1 \ll 1$	2
SL H2O Cleanup	$1 \ll 2$	4
Cum Self	$1 \ll 3$	8
Allow In Pussy	$1 \ll 4$	16
Hard mode	$1 \ll 5$	32
Verbose Messages	$1 \ll 6$	64
Cum Glow	$1 \ll 7$	128
Cum Texture Animation	$1 \ll 8$	256
Use Face Animations	$1 \ll 9$	512
Use Project Arousal to limit cleanup	$1 \ll 10$	1024
Male Only	$1 \ll 11$	2048
Face 1 Used	$1 \ll 12$	4096
Face 2 Used	$1 \ll 13$	8192
Face 3 Used	$1 \ll 14$	16384
Face 4 Used	$1 \ll 15$	32768
RLV Active	$1 \ll 16$	65536
System Menu Disabled	$1 \ll 17$	131072
API Menu Disabled	$1 \ll 18$	262144
Cleanup Disabled	$1 \ll 19$	524288
Pussy is plugged	$1 \ll 20$	1048575
Ass is plugged	$1 \ll 21$	2097152
Face 5 Used	$1 \ll 22$	4194304
System Lockout Active	$1 \ll 23$	8388608
Hands Add-on Present	$1 \ll 24$	16777216
Deciduan Used	$1 \ll 25$	33554432
Alt Face Accy	$1 \ll 26$	67108864
Project Arousal Used	$1 \ll 27$	134217728
No longer used	$1 \ll 28$	268435456
Drip Down Effect	$1 \ll 29$	536870912
Statistic Tracking	$1 \ll 30$	1073741824
She-male option	$1 \ll 31$	-2147483648

To determine if a bit is set (turned on) you would simply test the system\_settings value against the bit you wanted to check:

```
If (system_settings & (1<<16)) {  
    /*If the test returns true, bit 16 is turned on and would indicate that the user had RLV turned on in their viewer.*/
```

```
}
```

```
else {
```

```
    /*Bit 16 is turned off and the user is not using RLV*/
```

```
}
```

Or

```
If (system_settings & 65536) {
```

```
    /*This test is functionally the same as 'if (system_settings & (1<<16))'*/
```

```
}
```

```
else {
```

```
    /*Bit 16 is turned off and the user is not using RLV*/
```

```
}
```

## 14. Appendix F

### Additional Commands added in API v1.8

#### *Enable the System Lock*

This command will prevent the wearer, if using RLV, from detaching their INM system or making any changes to the system settings in their INM system for the duration of the lock. This lock resets whenever the system is attached or the user logs in wearing their INM system.

*Linked message to send to API:*

```
//MessageLinked(LINK_THIS/SET, 2550, "SystemLock/" + "0 or 1", <target_av_key>);
```

*Target\_av\_key* is the UUID (key) for owner of the cum system you want to lock.

**Note:**

Regardless of the key sent, *<target\_av\_key>* will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

## **Disable Cleanup**

This command will prevent the wearer from cleaning themselves in Linden water, any INM capable cleaning item such as showers, tubs, etc, or by using one of the Get it Off! towels. Another person may still clean them using a Get it Off! Towel however. This lock resets whenever the system is attached or the user logs in wearing their INM system.

*Linked message to send to API:*

```
//MessageLinked(LINK_THIS/SET, 2550, "DisableCleanUp|" + "0 or 1",  
<target_av_key>);
```

*Target\_av\_key* is the UUID (key) for owner of the cum system you want to disable cleanup on.

**Note:**

Regardless of the key sent, *<target\_av\_key>* will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

### **Disable functions requiring the use of hands**

This command will prevent the wearer from accessing any functions that require the use of their hands. This includes wiping their face and licking their arms, hands, or breasts. This value defaults to 1 whenever the system is attached or the user logs in wearing their INM system.

***Linked message to send to API:***

```
!!MessageLinked(LINK_THIS/SET, 2550, "HandFuncs|" + "0 or 1", <target_av_key>);
```

Target\_av\_key is the UUID (key) for owner of the cum system you want to disable the wearers ability to use their hands.

**Note:**

Regardless of the key sent, <target\_av\_key> will revert to the key for the owner of the object the API script is in if the object is a worn attachment!

## **Plug Pussy and Plug Ass**

These commands will stop any vaginal or anal drips that may be ongoing in a users INM system, prevent cleanup of those areas, and prevent another user from cumming in either location.

*Linked message to send to API:*

```
//MessageLinked(LINK_THIS/SET, 2550 , "PussyPlugged|" + "0 or 1",  
<target_av_key>);
```

```
//MessageLinked(LINK_THIS/SET, 2550 , "AssPlugged|" + "0 or 1", <target_av_key>);
```

Target\_av\_key is the UUID (key) for owner of the cum system you want to disable the wearers ability to use their hands.

**Note:**

Regardless of the key sent, <target\_av\_key> will revert to the key for the owner of the object the API script is in if the object is a worn attachment!