VR authentication through 3D key block building

Romain Fournier, Benjamin Freeling, Miguel Gervilla, Martin Heitz, Paul Viville, Kévin Berenger, Flavien Lecuyer, Antonio Capobianco

VR authentication through 3D key block building

Romain Fournier [1] Benjamin Freeling [1] Miguel Gervilla [1] Martin Heitz [2,1] Paul Viville [1]
Kévin Bérenger [1] Flavien Lécuyer [1] Antonio Capobianco [1]

[1] ICube, Université Strasbourg
[2] INETUM

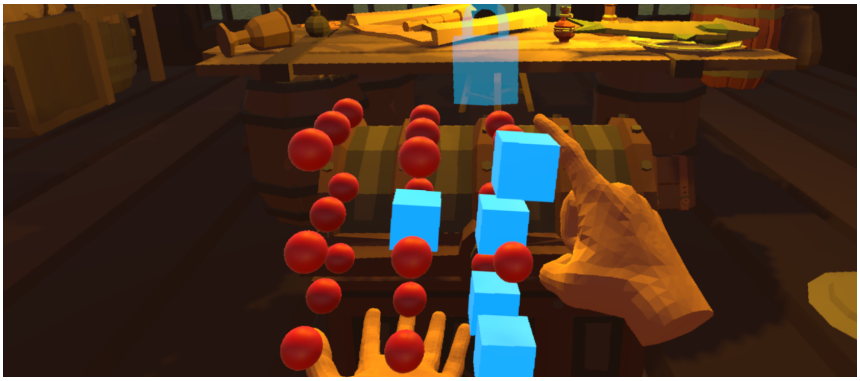

Figure 1: Our cube interface

ABSTRACT

With the rise of the metaverse, more and more massively multi-user applications are to be expected. Because some of the data integrated in those virtual environments can be seen as highly sensitive, it is important to reinforce the applications with authentication methods to keep them safe. However, and while there are many authentication methods for 2D applications that could be regarded as safe enough, most of them do not provide a sufficient ease of use for VR applications.

Henceforth, we propose an authentication method adapted to 3D user interfaces, based on toggling areas in a cubic grid. As illustrated in Figure 1, this leads to the construction of a voxel-based volume, which is an object simple enough to remember and playful to build, yet difficult to guess thanks to a high number of possibilities. For a more portable authentication method, the cubic grid is complemented by the generation of virtual keys representing the access a user has to a certain lock. The key acts as a metaphor that allows to easily share or grant access to secured objects or places in the VE.

Index Terms: Human-centered computing—Virtual reality; Security and privacy—Usability in security and privacy

1 INTRODUCTION

Virtual reality applications tend to include more and more users, raising the question of data security in the virtual environments and the easy management of access and rights in a social environment such as the Metaverse. While there are many families of authentication methods that could be used in 3D user interfaces, many of them have too important drawbacks to be pertinent; biometrics are incompatible with many devices; and alphanumeric passwords are usually defined as simple by users, reducing their usefulness for security purposes [1].

For a kind of passwords that can be both simple to remember and easily usable in 3D, we propose a 3D cube with voxels to switch on and off. The use of a such an object makes it easier to remember more complex data, thanks to the 3D structure [2]. Furthermore, the complexity of the authentication method can be adapted according to the desired security level. The ease of use and safety is also improved by the generation of a key for more portability, and randomization and obfuscation to prevent other users to guess the password through sheer observation.

We also propose a management interface based on a key metaphor that allows to easily share and manage access to items and places in VR. Indeed, contrary to identification techniques for personal items such as smartphones, social immersive environments also raise the problem of sharing access with other users.

2 AUTHENTICATION PROCEDURE

The main entry point of our interface is a cubic grid, in which the user can toggle the voxels state, switching them either on or off. While it has been demonstrated that such an authentication method might lead to lower usability if has a higher degree of security when compared to other information-based methods such as pattern lock or PIN system [3]. For a simple to remember – and to input – password while ensuring strong security, we defined the base grid as a cubic grid of $3 \times 3 \times 3$ voxels. Indeed, such an object can be efficiently memorized, and the complexity of the password obtained with such a grid is as follows: $2\ states^{3\times3\times3\ voxels} = 2^{27} \approx 134\ million$.

Of course, one of the advantages of such an interface is its flexibility so that the complexity of the password can be adjusted to the needs of the context. Indeed, the shape of the grid could for instance be changed to a sphere or a pyramid, and the size of the shape can also be altered depending on the number of possibilities any context asks for. Another lever for adjustment is the number of possible states for each voxel. While we show here the use of our method with binary inputs, the switched on voxels can easily take different appearances, allowing for several states represented by a color or texture.

Whenever the user tries to interact with a locked object without possessing the key, the cube interface appears. The interface also

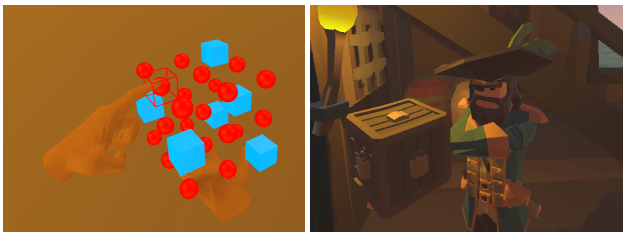

Figure 2: The obfuscated cube as seen by the user (left) and by an observer (right)

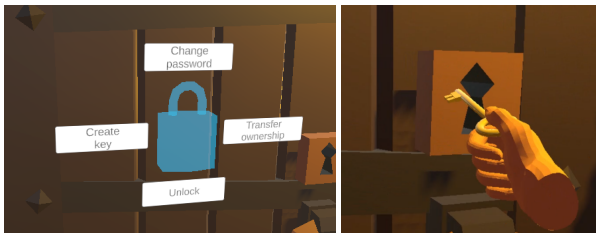

Figure 3: Management interface (left) and key (right)

appears when the user wants to enter the password management interface, as a safety measure.

For this interaction, we consider the hand used for interacting with the lock as the dominant hand. The cube interface appears placed on the other hand. The activation of the voxels is done with the dominant hand upon the trigger release.

The cube interface is further enhanced by two safety measures: a randomization of the initial cube state, and an obfuscation for the other users. The randomization of the initial state is done by switching on between 0 and 5 voxels chosen at random. Thanks to this, the movements done by a user to input their code change at each input, hindering the attempts of guessing the password through sheer observation. The observation is further made more difficult by the use of an obfuscation skin for the cube: instead of seeing the voxels, the observing see an opaque cube, allowing them to know that the user is inputting a code but not to know what the code is. An example of what an observer may see is shown in Figure 2.

3 Key metaphor and access management

For more portability once the user is authenticated, a key is generated and serves as the authentication modality. For more ease of use, the key is represented by a "super-key" containing all the user's current keys. It appears only near a compatible lock and is stored once used.

When the user approaches their key to a lock, the systems checks if the user possesses a valid key for the lock. This way, if the user does have a correct key, the lock is automatically opened when the super-key is used on the lock.

By interacting with the lock, the owner of an object can open the management interface (Figure 3) to handle password administration and sharing access. First, the user is prompted to input the password, revealing 4 interactions to manage the lock:

Share the lock with another user, by creating a copy of the key for another user. This shared key will grant the other user access to the locked object, while keeping them from altering the code, still used for the super-user access.

Password modification to reset the lock with a new password. If the password is changed to a different one, the shared accesses will also automatically be revoked. It is to note that the users having a shared key will not be notified of this, unless they try to open the lock, as a way to prevent social engineering.

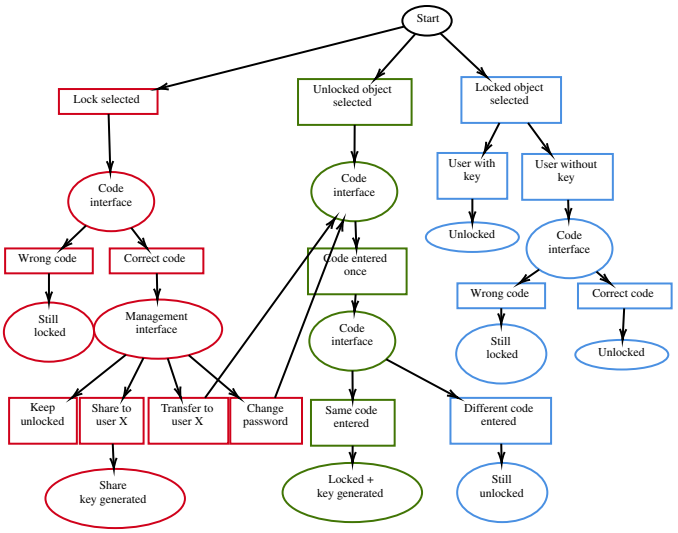

Figure 4: Interface workflow

Transfer the property of the lock to another user. This action deletes all keys previously created for the lock, and prompts the new owner to generate a new password and key.

Unlock the object until the owner decides otherwise. By default, any unlocked object will be closed and locked anew a few seconds after it is unlocked. By using the unlock option, it will instead stay opened for everyone until further notice.

For the **share** and **tranfer** actions, another user is involved. The selection of the second user is to be done by pointing them directly, needing for both the owner of the lock and the second user to present at the same time near the lock. This ensures a more secure action, as the action is performed by the presence of both users instead of only one of them.

4 Conclusion

In this paper, we propose a novel authentication interface based on the paradigm of filling voxels on the 3D cube. This method has the advantage of being scalable: depending on the number of voxels to input, the passwords defined with this method can either tend on the side of simplicity, or on the security one. The password input is also complemented by multiple management functions, all provided in a single spot for simplicity sake. Furthermore, the safety of the passwords is improved by an obfuscated view for potential observers, and a randomization of the cube before input. The password interface is combined with the use of a visual object, a key, as a token for more advances actions such as sharing the lock or reopening an owned lock, making for a more portable authentifier.

References

[1] F. A. Alsulaiman and A. El Saddik. Three-dimensional password for more secure authentication. *IEEE Transactions on Instrumentation and Measurement*, 57(9):1929–1938, Sep. 2008. doi: 10.1109/TIM.2008.919905

[2] M. Tavanti and M. Lind. 2d vs 3d, implications on spatial memory. In *IEEE Symposium on Information Visualization, 2001. INFOVIS 2001.*, pp. 139–145. IEEE, 2001.

[3] Z. Yu, H.-N. Liang, C. Fleming, and K. L. Man. An exploration of usable authentication mechanisms for virtual reality systems. In *2016 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, pp. 458–460. IEEE, 2016.