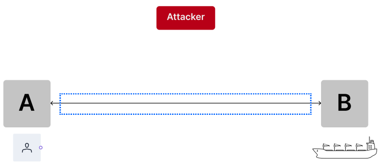
DISPEL

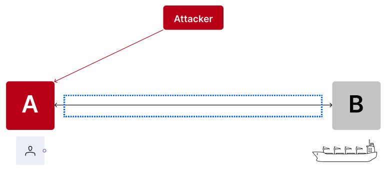
How to Explain Low-Attribution Networks to Your Colleagues



There are two common misconceptions that have driven much of network security over the past thirty years.

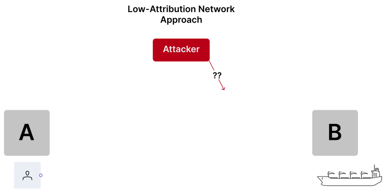
1. Attackers care most about the content traveling between A and B, and
2. The best way to protect A and B is to build thicker, stronger static walls around them.

To combat the first “attacker axiom,” traffic has long been encrypted, with ever more complicated keys and ciphers. In fact, defenders have stayed ahead in this arms race so long that attackers have changed their strategy.



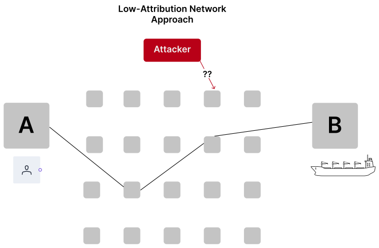
Modern attackers now overlook the content of a given transmission and instead focus on where both of the endpoints are. After a period of reconnaissance, most malicious actors become aware that B is a high value target like a warship, and that A is a much easier target to compromise. In this case, the attacker will simply pivot horizontally, and use A to get to B.

The standard defensive response is to build additional perimeter defenses around A and B to protect them. We believe that is simply not good enough, if the attacker knows where A and B are, they will find a way through those static walls — given the benefit of time.



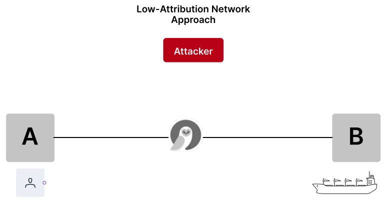
Instead, a better way to protect A and B is to disassociate them entirely. Without the connection between them, an attacker will not know where to start.

To dissassociate A from B, you need to deploy what are known as Moving Target Defense or Low-Attribution Networks.



By passing traffic through a low-attribution network layer, the relationship between A and B becomes hidden from outside view, and the locations of A and B become hidden from each other.

In other words, an attacker cannot see that A and B are communicating with one another or where A and B are located. Thus, it becomes increasingly difficult to launch an attack against.



With Dispel, the creation of low-attribution networks is automated. The low-attribution network layer does not interrupt the user experience — A can access B rapidly.